

2009

Bachelorscriptie Logistieke Econometrie

Auteur: Irma Tempelaar (303495)

Begeleider: Wilco van den Heuvel

[ROOSTEREN IN TWEE FASES]

NAAR EEN OPTIMAAL ROOSTER VOOR SCHOLIEREN DOOR
MIDDEL VAN HET MINIMALISEREN VAN TUSSENUREN.

Inhoudsopgave

1. Voorwoord	3
2. Samenvatting.....	4
3. Inleiding en Probleemstelling.....	4
3.1. Algemene aspecten van een schoolrooster	4
3.2. Specifieke eigenschappen van dit schoolrooster	4
4. Methodologie	5
4.1. Inleiding	5
4.2. Fase 1.....	6
4.2.1. Heuristiek	6
4.2.2. Maximum koppelingen.....	7
4.3. Fase 2.....	7
4.3.1. Heuristiek	7
4.3.2. Exact algoritme	10
5. Resultaten	14
5.1. Datatransformatie	14
5.2. Data-analyse	15
5.3. Fase 1.....	15
5.3.1. Heuristiek	15
5.3.2. Maximum koppelingen.....	15
5.4. Fase 2.....	16
5.4.1. Heuristiek	16
5.4.2. Exact algoritme	18
6. Conclusie	21
7. Literatuurlijst	22
8. Bijlagen	23

1. Voorwoord

Al tijdens mijn opleiding op de middelbare school leek het mij leuk om een rooster te maken. Mijn klasgenoten en ik hadden veel tussenuren die vervolgens, met een beetje schuiven, helemaal niet nodig bleken. Toen ik hoorde dat je dit met Econometrie kon oplossen was mijn onderwerp voor de bachelorscriptie duidelijk! Toen het zo ver was dat de bachelorscriptie geschreven kon worden ben ik eerst terug geweest naar mijn middelbare school: het Stedelijk Dalton Lyceum te Dordrecht. Daar heb ik inzicht gekregen in de huidige planningswijze. Ook heb ik van hen de benodigde data gekregen om een realistisch roosterprobleem te creëren. Zo zal ik de onderbouw, zoals deze nu daadwerkelijk actief is, trachten te roosteren.

Het Stedelijk Dalton Lyceum combineert twee rooster programma's, Zermelo en Untis. Ik heb ook contact gehad met deze software bedrijven. Hen heb ik gevraagd naar een korte toelichting op hun software.

Namens Zermelo kon Bernard van Kesteren mij melden dat zij gebruik maken van allerlei benaderingen. Zo maken zij soms direct of indirect gebruik van 'graph colouring', waar ik ook een versie van zal gaan toepassen. Hij geeft aan dit soms heel goed werkt en soms heel slecht. "Als je denkt dat je "de" oplossing hebt, faalt die weer jammerlijk in een andere situatie." Daarom wordt er gebruik gemaakt van het mixen van benaderingen en een aantal dingen gebeurt ook random.

Mevrouw Brands kon namens Untis een andere benadering toelichten. Ze werken daar met een strafpuntensysteem. In principe kan een les namelijk op elk moment van de week gegeven worden, maar er zijn beperkingen (bijvoorbeeld een docent die op vrijdag vrij is). Deze les krijgt dan strafpunten. Aan de hand van strafpunten wordt de moeilijkste les eerst geplaatst, dan de iets minder moeilijke, enzovoort. Tenslotte wordt er nog geprobeerd of er door ruilen meer of betere mogelijkheden zijn om meer lessen te plaatsen.

Ik wil graag gebruik maken van deze gelegenheid om een aantal mensen te bedanken voor hun hulp bij het realiseren van deze scriptie. In de eerste plaats mijn scriptie begeleider, Wilco van den Heuvel, voor het voorzien van scherp commentaar en zeer nuttige tips. Uiteraard wil ik ook het Stedelijk Dalton Lyceum bedanken voor het kijkje achter de schermen en het mogen gebruiken van de data. Ten slotte wil ik ook mijn ouders en vriend bedanken voor de ondersteuning tijdens het schrijven en het voorzien van commentaar als leken op dit gebied, dat heeft voor veel helderheid heeft gezorgd.

Dan wil ik u tot slot veel leesplezier wensen.

2. Samenvatting

In deze scriptie is een rooster gemaakt voor een middelbare school. Dit is gedaan in twee opeenvolgende fases, waarbij in de eerste fase een lijnkleuringsprobleem opgelost is en in de tweede fase een plaatsingsprobleem. Het doel hierbij is om voor klassen roosters te genereren zonder tussenuren. Hierbij zijn verschillende cases behandeld met eisen ten aanzien van het rooster. Het is gelukt een rooster te genereren zonder tussenuren voor klassen.

3. Inleiding en Probleemstelling

In deze scriptie zal een rooster gegenereerd worden voor een middelbare school. Het roosterprobleem is in de wetenschap nog niet volledig opgelost. Het is dan ook te ambitieus om alle facetten van een rooster mee te nemen in deze scriptie. In dit hoofdstuk zal ik daarom verder toelichten met welke aspecten ik wel en met welke ik geen rekening houd. Deze aspecten zijn op te splitsen in algemene en specifieke aspecten. Deze zullen in de desbetreffende paragrafen hieronder besproken worden.

3.1. Algemene aspecten van een schoolrooster

Het allerbelangrijkste aan een rooster voor de middelbare school is dat in een lesuur er geen docent of klas dubbel geroosterd is. Dit is de absolute basis van het rooster, dus hier zal uiteraard rekening mee worden gehouden.

Bij het maken van een rooster dient er bepaald te worden welke docenten welke klassen welke vakken gaan geven. Deze indeling wordt als gegeven beschouwd op basis van de indeling zoals deze al door het Stedelijk Dalton Lyceum is gemaakt.

Een andere algemene eigenschap van een rooster op een middelbare school is de zaalcapaciteit. Zo is het logisch dat er specifieke lokalen ingericht zijn voor tekenen of gym en dat er dus niet meer klassen tegelijk dit vak kunnen hebben dan dat er specifieke lokalen zijn. Hier zal in deze scriptie geen rekening mee gehouden worden.

In de praktijk is het zo dat docenten op sommige dagen niet kunnen of willen werken. Als hier toestemming voor gegeven is door het management heeft de roostermaker hier rekening mee te houden. Ik zal hier geen rekening mee houden, omdat ik geen beschikking heb over deze gegevens.

3.2. Specifieke eigenschappen van dit schoolrooster

Een zeer specifieke eigenschap van een Daltonschoon is het Daltonuur. Hier zal ik dan ook rekening mee houden en vaak apart naar verwijzen, daarom nu een korte toelichting. Het idee achter een Daltonschoon is dat scholieren zelf moeten leren plannen. Hiervoor is het Daltonuur in het leven geroepen. De scholieren krijgen verschillende Daltontaken, het aantal en de frequentie is niet van belang voor het rooster. Deze taken kunnen van alles inhouden, van een stencil met vragen invullen tot een tekening maken of een werkstuk en alles ertussen. Deze moeten af zijn op het moment dat de nieuwe Daltontaken uitgedeeld worden. Hiervoor wordt verder geen planning aangereikt. Scholieren hebben het vijfde uur Daltonuur. In dit uur mogen ze zelf weten bij welke docent ze gaan zitten en welk vak of welke Daltontaak ze gaan doen. Ze mogen ook huiswerk maken of extra uitleg vragen. Hierdoor worden ze gedwongen een planning te maken en overzicht te houden.

Een andere specifieke eigenschap is dat op donderdag de laatste drie uur, van de tien, vrij moeten zijn voor alle klassen. Dit is omdat het dan CSM-middag is. CSM staat voor Cultuur, Sport en Maatschappij. Hierbij doen de scholieren verschillende activiteiten die binnen deze drie categorieën vallen. Denk hierbij aan een softbal toernooi of vrijwilligerswerk. Er zullen in de scriptie een aantal verschillende roosters gemaakt worden op basis van andere aspecten die van belang zijn. Hier zal deze eigenschap ook in voorkomen.

Verder is het zo dat de docenten op dinsdag de laatste drie uur willen vergaderen. Hierdoor mogen de klassen op deze tijdstippen ook geen les hebben. Deze eigenschap zal op dezelfde manier als de CSM-middag verwerkt worden.

Er zijn vakken die een klas maar een half jaar krijgt en daarna ingewisseld worden voor een ander vak. Denk aan muziek en drama. De helft van de klassen heeft muziek de eerste helft van het jaar en drama de tweede helft van het jaar en de andere klassen hebben dit precies andersom. Ik zal een rooster maken voor een halfjaar. Hierbij zal ik de lessen roosteren zoals deze gepland stonden voor de tweede helft van het schooljaar (2008-2009).

Ten slotte is het zo dat een klas maximaal twee uur per dag hetzelfde vak mag hebben. Daar komt bij dat als ze twee uur dit vak hebben, het aaneengesloten moet zijn in de vorm van een blokuur. Een blokuur mag niet onderbroken worden door het Daltonuur, maar wel door de kleine of grote pauze. Ik zal eerst een rooster maken zonder deze eisen en daarna trachten deze eisen in het rooster te verwerken en nog steeds een optimaal rooster te maken.

4. Methodologie

4.1. Inleiding

Voor het maken van het rooster zal ik gebruik maken van twee opeenvolgende fases:

Fase 1: De benodigde lessen kunnen weergegeven worden door middel van een tweedelingsgraaf met in het ene part de klasknopen en in het andere part de docentknopen. Vervolgens is elke les een verbinding tussen deze knopen. Het doel hierbij is om te bepalen welke lessen tegelijk gegeven kunnen worden zonder dat er klassen (docenten) twee lessen moeten volgen (geven). Dit is te omschrijven als een lijnkleuringsprobleem. Een lijnkleuringsprobleem is een probleem waarbij de lijnen in een graaf allemaal een kleur moeten krijgen, zodat lijnen met een gemeenschappelijk uiteinde nooit dezelfde kleur hebben en dat er zo min mogelijk kleuren gebruikt worden. Dit zal uitgevoerd worden door een zelf opgezette heuristiek.

Fase 2: In deze fase is er de beschikking over een set kleuren die aangeeft welke lessen tegelijk geroosterd kunnen worden. Nu zullen deze kleuren verdeeld worden over de lessen in de week met als doel het minimaliseren van tussenuren voor de klassen. Dit zal met een zelf opgezette heuristiek en met een exact algoritme gedaan worden. Deze zullen met elkaar vergeleken worden.

Door de splitsing in fases is het gehele rooster proces een heuristiek geworden. De exacte algoritmes geven wellicht optimale oplossingen voor de deelproblemen, maar dit betekent nadrukkelijk niet dat het eindrooster optimaal is.

4.2.Fase 1

In deze eerste fase nemen we de toekenning van klassen aan docenten voor bepaalde vakken als gegeven. Tevens is bekend hoe vaak deze les in een week voorkomt. Het komt ook voor dat een klas meerdere vakken krijgt van een docent (denk bijvoorbeeld aan de mentor die mentoruur geeft, maar ook aardrijkskunde). In dit geval wordt er geen onderscheid gemaakt in welk vak er les gegeven wordt.

Het doel van deze fase is om met de gegevens zoals hierboven vermeld een verdeling te maken van welke lessen (combinaties van klassen en docenten) allemaal tegelijk gegeven kunnen worden zonder dat er klassen dubbele lessen volgen of docenten dubbele lessen geven.

In de volgende fase zal getracht worden een rooster te creëren met zo min mogelijk tussenuren voor de klassen. Dit, in combinatie met de veronderstelling dat er zo min mogelijk tussenuren ontstaan als er steeds zo veel mogelijk leerlingen les hebben, leidt er toe dat er in deze fase steeds zo veel mogelijk lessen aan een bepaalde kleur toegevoegd worden.

Dit kan door middel van een zelf opgestelde heuristiek of maximum koppelingen. Beide zullen in de desbetreffende paragraaf toegelicht worden.

4.2.1. Heuristiek

Dit leidt tot een zogenoemde 'greedy' heuristiek welke steeds de moeilijkste les in de eerst mogelijke kleur plaatst. Deze heuristiek is gebaseerd op ideeën uit het artikel van D. de Werra (1985). In deze scriptie wordt met de aanduiding 'moeilijkste les' de les bedoeld die het moeilijkst te roosteren is.

Er zijn verschillende manieren waarop de moeilijkste les geselecteerd kan worden. Ik hanteer de volgende aanpak: zoek eerst de klasknoop met de hoogste graad. Bepaal vervolgens van deze klas alle docenten waar ze nog les van moeten krijgen. Bekijk vervolgens voor deze docenten hoeveel lijnen zij nog hebben (met alle klassen) en selecteer de moeilijkste docent. Bij een gelijke stand zal willekeurig bepaald worden welke klas/leraar geselecteerd wordt op basis van de uniforme verdeling.

Dit leidt tot de volgende heuristiek:

- 1) Kies de moeilijkste klas, met andere woorden de klas met de meeste lijnen.
- 2) Selecteer de docenten die nog les moeten geven aan deze klas.
- 3) Selecteer uit de lijst van docenten uit stap 2 de moeilijkste docent, met andere woorden met de meeste lijnen naar alle klassen.
- 4) Wijs de eerst mogelijke kleur toe, met andere woorden de eerste kleur beschikbaar waar zowel de docent als de klas nog niet in voorkomen.
- 5) Laat de gekleurde lijn vanaf nu buiten beschouwing.
- 6) Indien er nog minimaal één ongekleurde lijn is, ga terug naar 1. Anders beëindigen.

Het aantal kleuren waarin dit algoritme resulteert is moeilijk van te voren te bepalen. Wel kan er opgemerkt worden dat het minimaal gelijk is aan de hoogste graad van alle klasknopen en docentknopen.

4.2.2. Maximum koppelingen

Indien de heuristiek van subparagraaf 4.2.1. niet in een acceptabele oplossing resulteert, zal er overgegaan worden op de methode van maximum koppelingen. Een koppeling is een verzameling lijnen in een graaf zó dat geen twee lijnen een uiteinde gemeen hebben. Een maximum koppeling is een koppeling met een zo groot mogelijk aantal lijnen. Deze maximum koppeling wordt samengesteld door middel van de Hongaarse methode. Voor het lijnkleuringsprobleem wordt elke maximum koppeling een kleur en vervolgens worden alle lijnen in de koppeling uit de graaf verwijderd. Hierna wordt opnieuw een maximum koppeling samengesteld enzovoort. Dit proces gaat door tot er geen ongekleurde lijnen meer over zijn.

Het is onduidelijk of dit proces uiteindelijk tot het minimale aantal kleuren zal leiden. Er is bewezen dat een tweedelingsgraaf waarin elk punt een graad k heeft, in k koppelingen te verdelen is. Op basis van deze stelling zou de graaf met de benodigde lessen uitgebreid kunnen worden met dummy klassen en dummy lessen, zodat elke knoop dezelfde graad krijgt. Een lijn tussen een klas of een docent met een dummy knoop staat gelijk aan geen les. De graad van elke knoop moet dan gelijk zijn aan de hoogste graad van alle klas en docentknoten in de initiële tweedelingsgraaf. Het is moeilijk vooraf te bepalen of het mogelijk is om deze situatie te verkrijgen.

4.3.Fase 2

In deze tweede fase nemen we de indeling van bepaalde lessen in kleuren zoals samengesteld in fase 1 als gegeven. Het doel van deze fase is om met deze kleuren een rooster te genereren met zo min mogelijk tussenuren voor de klassen.

Een tussenuur is als volgt gedefinieerd:

Een tussenuur voor een klas is een lesuur zonder een geroosterd vak, maar waarbij in ten minste één van de uren voor en na dat lesuur wel een vak geroosterd is.

Zoals in de algemene inleiding ook al kort uiteengezet is, wordt getracht deze fase door middel van een zelf opgezet algoritme en een exacte aanpak door middel van een model op te lossen. De methodes zullen in de desbetreffende subparagrafen nader toegelicht worden.

4.3.1. Heuristiek

Deze heuristiek begint met een model met betrekking tot het aantal betrokken partijen in lessen. Met de term 'aantal betrokken partijen' wordt in deze scriptie bedoeld het aantal klassen dat les heeft op een bepaald moment.

De doelstelling van het initiële model is om zo min mogelijk verschil te hebben in het aantal betrokken partijen tussen twee opeenvolgende lessen op een dag. In het Daltonuur zijn alle klassen aanwezig, het idee is dat rondom het Daltonuur de meeste partijen actief zijn en naar de uiteindes van de dag steeds minder. Dit zal tot gevolg moeten hebben dat er weinig tussenuren ontstaan. Voor het model worden er vijf kleuren achteraan toegevoegd aan de n kleuren zoals gevonden in fase 1 welke het Daltonuur weergeven. Dus de kleuren $n + 1$ tot en met $n + 5$ stellen de Daltonuren voor.

Model betrokken partijen

INDEXDOMEINEN

$J :=$ Set kleuren $j \in \{1, 2, \dots, n-1, n, n+1, \dots, n+5\}$
 $T :=$ Set lesuren op een dag $t \in \{0, 1, 2, \dots, 10, 11\}$
 $D :=$ Set dagen $d \in \{\text{maandag, dinsdag, } \dots, \text{vrijdag}\}$

PARAMETERS

$p_j =$ Aantal betrokken partijen in kleur j

$r =$ Minimum aantal betrokken partijen per dag $\left(0 \leq r \leq \left\lfloor \frac{\sum_{j \in J} p_j}{|D|} \right\rfloor\right)$

VARIABELEN

$z_{j,t,d} = \begin{cases} 1 & \text{als kleur } j \text{ in lesuur } t \text{ op dag } d \text{ wordt geplaatst} \\ 0 & \text{anders} \end{cases}$

$vp_{t,d} =$ Verschil tussen het aantal betrokken partijen tussen tijdstip t en $t-1$ op dag d .

DOELFUNCTIE

Min $\sum_{t \in T} \sum_{d \in D} vp_{t,d}$

RESTRICTIES

- (1) $vp_{t,d} \geq \sum_{j \in J} p_j \cdot z_{j,t,d} - \sum_{j \in J} p_j \cdot z_{j,t-1,d} \quad \forall t \in T \setminus \{0\}, d \in D$
- (2) $vp_{t,d} \geq \sum_{j \in J} p_j \cdot z_{j,t-1,d} - \sum_{j \in J} p_j \cdot z_{j,t,d} \quad \forall t \in T \setminus \{0\}, d \in D$
- (3) $\sum_{t \in T} \sum_{d \in D} z_{j,t,d} = 1 \quad \forall j \in J$
- (4) $\sum_{j \in J} z_{j,t,d} \leq 1 \quad \forall t \in T, d \in D$
- (5) $z_{j,0,d} = 0 \quad \forall j \in J, d \in D$
- (6) $z_{j,11,d} = 0 \quad \forall j \in J, d \in D$
- (7) $\sum_{d \in D} \sum_{j=n}^{n+5} z_{j,5,d} = 5$
- (8) $\sum_{j \in J} \sum_{t \in T} p_j \cdot z_{j,t,d} \geq r \quad \forall d \in D$
- (9) $z_{j,t,d} \in \{0, 1\} \quad \forall j \in J, t \in T, d \in D$
- (10) $vp_{t,d} \geq 0 \quad \forall t \in T, d \in D$

TOELICHTING

Er is een fictief lesuur aan het begin en aan het eind van de dag toegevoegd. Dit met het oog op de eigenlijke doelstelling: minimaliseren van het aantal tussenuren voor klassen. Anders zou er namelijk een situatie kunnen ontstaan dat er op de eerste en laatste lesuren altijd een les geplaatst wordt. Dit is zonder deze fictieve lesuren namelijk lucratief, omdat er alleen een verschil is voor of na het lesuur. Bij alle andere lesuren zijn er voor én na het lesuur verschillen. Het is uiteraard ongewenst om altijd het eerste en laatste lesuur van de dag te roosteren, omdat het rooster zo compact mogelijk gemaakt moet worden voor het minimaliseren van tussenuren.

Voor de juiste waarde van r geldt dat er gebruik gemaakt dient te worden van een proces van ‘trial and error’. Er zullen binnen de grenzen hoge waarden zijn, die wel toegestaan zijn, maar waarvoor het heel lang duurt om de juiste oplossing te vinden. Er zullen binnen de grenzen ook lage waarden zijn, die wel toegestaan zijn, maar dan komt je redelijke verdeling weer in het gedrang. Ergens hier tussen zit de juiste r .

- (1) & (2) Per definitie geeft $vp_{t,d}$ het absolute verschil aan tussen het aantal betrokken partijen in twee opeenvolgende uren. Omdat een absolute waarde nemen geen lineaire functie is moet deze waarde op een andere manier toegekend worden. Deze restricties zorgen er samen voor dat de variabelen de juiste waarde aannemen, omdat deze het verschil zowel positief als negatief weergeven en de variabelen groter dan of gelijk aan allebei moeten zijn. Aan de andere kant worden de variabelen geminimaliseerd wat ervoor zorgt dat de variabelen de waarde krijgen van het positieve verschil.

Ter illustratie de drie mogelijke situaties nader bekeken

- I. In lesuur t zijn meer partijen, zeg a , betrokken dan in lesuur $t - 1$.
Restrictie (1) eist dat $vp_{t,d} \geq a$ en restrictie (2) eist dat $vp_{t,d} \geq -a$. Deze gecombineerd zorgt ervoor dat $vp_{t,d} \geq a$ geldt. Door het minimaliseren van de som over de variabele zal een zo laag mogelijke waarde gekozen worden, wat dus a zal zijn. Dat is precies het verschil in betrokken partijen tussen de twee opeenvolgende uren.
 - II. In lesuur t zijn minder partijen, zeg a , betrokken dan in lesuur $t - 1$.
Restrictie (1) eist dat $vp_{t,d} \geq -a$ en restrictie (2) eist dat $vp_{t,d} \geq a$. Via dezelfde redenering als bij I. volgt hieruit dat de variabele de waarde a krijgt, precies het verschil in betrokken partijen tussen de twee opeenvolgende uren.
 - III. In lesuur t zijn evenveel partijen betrokken als in lesuur $t - 1$.
Restrictie (1) eist dat $vp_{t,d} \geq 0$ en restrictie (2) eist dat $vp_{t,d} \geq 0$. Deze gecombineerd zorgt ervoor dat $vp_{t,d} \geq 0$ geldt. Via een soort gelijke redenering als bij I. en II. volgt dat de variabele 0 wordt. Dat is precies het verschil in betrokken partijen tussen de twee opeenvolgende uren.
- (3) Uiteraard moeten alle lessen die gegeven moeten worden ook geplaatst worden in een week. Daarom moet elke kleur precies 1 keer geplaatst worden in een week.
- (4) De kleuren zijn zo ingedeeld dat deze lessen op hetzelfde tijdstip gegeven kunnen worden. Het is dus niet toegestaan om meerdere kleuren op één tijdstip te plaatsen, want dan krijgen we overlap en dat is niet toegestaan. Het is echter wel toegestaan om een lesuur leeg te laten.
- (5) & (6) Deze restricties geven aan dat de fictieve lessen geen kleur mogen bevatten.
- (7) De eerste vijf kleuren die toegevoegd zijn stonden voor de Daltonuren. Deze moeten elke dag op het vijfde uur geplaatst worden. Samen met de restrictie dat er slechts één kleur per lesuur geplaatst mag worden zorgt dit ervoor dat de Daltonuren geroosterd worden.
- (8) Het loont om zo min mogelijk overgangen te hebben tussen het aantal betrokken partijen tussen lessen en het zal dus rendabel zijn om hele drukke dagen te plannen met elk uur zoveel mogelijk betrokken partijen van het eerste tot het laatste uur. Dit heeft als gevolg dat er zeer waarschijnlijk een dag overblijft met heel weinig betrokken partijen. Dit is volgens het idee van de school niet wenselijk, omdat iedereen vijf dagen in de week aanwezig moet zijn. Hiervoor dient deze restrictie, zodat er een enigszins redelijke spreiding gecreëerd wordt.

(9) & (10) volgen per definitie.

De uitkomst van dit model zal gebruikt worden om roosters te genereren. Vervolgens zal er gekeken worden of er met behulp van verwisselen of verplaatsen een beter rooster gemaakt kan worden.

4.3.2. Exact algoritme

Voor de exacte methode zal gebruik gemaakt worden van twee modellen, waarbij het ene model een verdere uitbreiding is van het andere model. Het eerste model zal van de gegeven kleuren een rooster maken met zo min mogelijk tussenuren. Hierbij wordt geen rekening gehouden met welk vak er gegeven wordt. Zo kan het dus heel goed zijn dat in de optimale oplossing van dit model een klas vier uur achter elkaar wiskunde krijgt. Dit is niet zoals het Stedelijk Dalton Lyceum een rooster voor ogen ziet. Daarom zal het model uitgebreid worden met een bepaalde voorwaarde. Deze voorwaarde is dat een vak maximaal twee keer per dag gegeven mag worden, waarbij deze uren aansluitend moeten zijn (een blokuur). Het Stedelijk Dalton Lyceum roostert graag blokken, maar hier wordt nu geen preferentie aan gegeven.

Voor beide modellen zullen vier cases uitgevoerd worden:

Case 1: Alles mag vrij geroosterd worden.

Case 2: De donderdagmiddag moet vrij blijven voor CSM activiteiten.

Case 3: De dinsdagmiddag moet vrij blijven voor de vergadering van de docenten.

Case 4: De dinsdagmiddag moet vrij blijven voor de vergadering van de docenten en de donderdagmiddag moet vrij blijven voor CSM activiteiten.

Model voor een minimaal aantal tussenuren zonder verdere beperkingen

INDEXDOMEINEN

$I :=$ Set klassen	$i \in \{1, 2, \dots, m\}$
$J :=$ Set kleuren	$j \in \{1, 2, \dots, n\}$
$T :=$ Set lesuren op een dag	$t \in \{1, 2, 3, \dots, 10\}$
$D :=$ Set dagen	$d \in \{\text{maandag, dinsdag, } \dots, \text{vrijdag}\}$

PARAMETERS

$$y_{i,j} = \begin{cases} 1 & \text{als klas } i \text{ les heeft in kleur } j \\ 0 & \text{anders} \end{cases}$$

VARIABELEN

$$x_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft tijdens lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$z_{j,t,d} = \begin{cases} 1 & \text{als kleur } j \text{ geplaatst is tijdens lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$v_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ op lesuur } t \text{ op dag } d \text{ een tussenuur heeft} \\ 0 & \text{anders} \end{cases}$$

$$e_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft op of voor lesuur } t \text{ op dag } d \text{ (Dus als } \sum_{\tau=1}^t x_{i,\tau,d} > 0) \\ 0 & \text{anders} \end{cases}$$

$$l_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft op of na lesuur } t \text{ op dag } d \text{ (Dus als } \sum_{\tau=t}^{10} x_{i,\tau,d} > 0) \\ 0 & \text{anders} \end{cases}$$

DOELFUNCTIE

$$\min \sum_{i \in I} \sum_{t \in T} \sum_{d \in D} v_{i,t,d}$$

De doelstelling is om zo min mogelijk tussenuren te hebben. Hierbij wordt geen onderscheid gemaakt tussen de verdeling van het aantal tussenuren over de klassen. Daarom kunnen de tussenuren over alle klassen, lesuren en dagen gesommeerd worden.

RESTRICTIES

- | | |
|--|---|
| (1) $x_{i,5,d} = 1$ | $\forall i \in I, d \in D$ |
| (2) $x_{i,t,d} = \sum_{j \in J} y_{i,j} z_{j,t,d}$ | $\forall i \in I, t \in T \setminus \{5\}, d \in D$ |
| (3) $\sum_{t \in T} \sum_{d \in D} z_{j,t,d} = 1$ | $\forall j \in J$ |
| (4) $\sum_{j \in J} z_{j,t,d} \leq 1$ | $\forall t \in T, d \in D$ |
| (5) $e_{i,t,d} \geq x_{i,t,d}$ | $\forall i \in I, t \in T, d \in D$ |
| (6) $e_{i,t,d} \geq e_{i,t-1,d}$ | $\forall i \in I, t \in T \setminus \{1\}, d \in D$ |
| (7) $l_{i,t,d} \geq x_{i,t,d}$ | $\forall i \in I, t \in T, d \in D$ |
| (8) $l_{i,t,d} \leq l_{i,t-1,d}$ | $\forall i \in I, t \in T \setminus \{1\}, d \in D$ |
| (9) $v_{i,t,d} \geq e_{i,t,d} + l_{i,t,d} - x_{i,t,d} - 1$ | $\forall i \in I, t \in T, d \in D$ |
| (10) $x_{i,t,d}, z_{j,t,d}, e_{i,t,d}, l_{i,t,d}, v_{i,t,d} \in \{0,1\}$ | $\forall i \in I, t \in T, d \in D, j \in J$ |

TOELICHTING

- (1) Het vijfde uur is voor iedereen Daltonuur, alle klassen hebben dit uur dus les.
- (2) In alle uren buiten het Daltonuur heeft een klas les als en alleen als de klas in de kleur zit die op dat tijdstip geplaatst is.
- (3) Het grondrooster is uiteraard opgesteld met de gedachte dat deze lessen gegeven moeten worden. Daarom moeten alle kleuren geplaatst worden. Dit moet exact één keer gebeuren voor elke kleur.
- (4) De kleuren zijn zo ingedeeld dat deze lessen op hetzelfde tijdstip gegeven kunnen worden. Het is dus niet toegestaan om meerdere kleuren op één tijdstip te plaatsen, want dan krijgen we overlap en dat is niet toegestaan. Het lesuur mag echter wel vrij blijven.
- (5) Dit volgt uit de definitie, omdat een klas op of voor een lesuur les heeft als de klas les heeft in dat lesuur. Indien een klas geen les heeft in een lesuur kan het nog steeds zo zijn dat er daarvoor les geweest is, daarom is het een ongelijkheidsrestrictie en geen gelijkheidsrestrictie.
- (6) Deze restrictie zorgt voor de consistentie van de variabele die aangeeft of een klas op of voor een bepaald tijdstip op een dag les heeft. Als er namelijk eenmaal een lesuur is waarop een klas op een dag op of voor dat tijdstip een les heeft, zal die klas alle uren daarna ook in ieder geval een les voor dat lesuur hebben.

- (7) Dit volgt uit de definitie, omdat een klas op of na een lesuur les heeft als de klas les heeft in dat lesuur. Indien een klas geen les heeft in een lesuur kan het nog steeds zo zijn dat er daarna nog een les geroosterd is, daarom is het een ongelijkheidsrestrictie en geen gelijkheidsrestrictie.
- (8) Deze restrictie zorgt voor de consistentie van de variabele die aangeeft of een klas op of na een bepaald tijdstip op een dag les heeft. Als er namelijk eenmaal een lesuur is waarop een klas op een dag op of na dat tijdstip een les heeft, zal die klas alle uren daarvoor ook in ieder geval een les na dat lesuur hebben.
- (9) Als je een tussenuur hebt, heb je les voor dat uur en na dat uur, maar niet op dat uur. Dan is $e_{i,t,d} + l_{i,t,d} - x_{i,t,d}$ gelijk aan 2. Je wilt dat $v_{i,t,d}$ gelijk aan 1 wordt dus $v_{i,t,d} \geq e_{i,t,d} + l_{i,t,d} - x_{i,t,d} - 1$ zorgt ervoor dat $v_{i,t,d} \geq 1$. Het is een binaire variabele dus $v_{i,t,d}$ wordt dan 1. Als er geen tussenuur is mag $v_{i,t,d}$ ook 1 worden, maar je minimaliseert deze som van $v_{i,t,d}$ over alle indexdomeinen dus dat zorgt er automatisch voor dat het dan een 0 wordt.
- (10) Alle variabelen zijn binair.

De combinatie van restrictie (4) en (5) geeft de juiste reeks voor $e_{i,t,d}$. Dit komt omdat eerst alle $e_{i,t,d}$'s 1 gemaakt worden als $x_{i,t,d}$ gelijk is aan 1 door restrictie (4). Daarna zorgt restrictie (5) dat alle $e_{i,t,d}$'s na de eerste $e_{i,t,d}$ gelijk aan 1 worden. Alle noodzakelijke enen zijn dan ingevuld. De rest moet een 0 worden en dit gebeurt doordat $e_{i,t,d}$ indirect door $v_{i,t,d}$ al in de doelfunctie opgenomen is en indien mogelijk zal $e_{i,t,d}$ dus altijd nul worden. Het zou voor kunnen komen dat de variabele niet meer aan de originele definitie voldoet, maar dit is voor de oplossing irrelevant.

De combinatie van (6) en (7) geeft met een gelijke redenatie de juiste reeks voor $l_{i,t,d}$.

Model voor een minimaal aantal tussenuren met bepaalde voorwaarde

INDEXDOMEINEN

$I :=$ Set klassen	$i \in \{1,2,3, \dots, m\}$
$J :=$ Set kleuren	$j \in \{1,2,3, \dots, n\}$
$T :=$ Set lesuren op een dag	$t \in \{0,1,2,3, \dots, 10,11\}$
$D :=$ Set dagen	$d \in \{\text{maandag, dinsdag, } \dots, \text{vrijdag}\}$
$C :=$ Set vakken	$c \in \{1,2,3, \dots, p\}$

PARAMETERS

$$y_{i,c,j} = \begin{cases} 1 & \text{als klas } i \text{ vak } c \text{ heeft in kleur } j \\ 0 & \text{anders} \end{cases}$$

VARIABELEN

$$x_{i,c,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft in vak } c \text{ tijdens lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$z_{j,t,d} = \begin{cases} 1 & \text{als kleur } j \text{ geplaatst is tijdens lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$v_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ op lesuur } t \text{ op dag } d \text{ een tussenuur heeft} \\ 0 & \text{anders} \end{cases}$$

$$e_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft op of voor lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$l_{i,t,d} = \begin{cases} 1 & \text{als klas } i \text{ les heeft op of na lesuur } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

$$o_{i,c,t,d} = \begin{cases} 1 & \text{als klas } i \text{ een overgang heeft tussen wel en niet les in vak } c \\ & \text{tussen tijdstip } t - 1 \text{ en } t \text{ op dag } d \\ 0 & \text{anders} \end{cases}$$

DOELFUNCTIE

$$\min \sum_{i \in I} \sum_{t \in T} \sum_{d \in D} v_{i,t,d}$$

RESTRICTIES

- | | |
|---|---|
| (1) $x_{i,4,5,d} = 1$ | $\forall i \in I, c \in C, d \in D$ |
| (2) $x_{i,c,0,d} = 0$ | $\forall i \in I, c \in C, d \in D$ |
| (3) $x_{i,c,11,d} = 0$ | $\forall i \in I, c \in C, d \in D$ |
| (4) $x_{i,c,t,d} = \sum_{j \in J} y_{i,c,j} z_{j,t,d}$ | $\forall i \in I, c \in C, t \in T \setminus \{0,5,11\}, d \in D$ |
| (5) $\sum_{t \in T} \sum_{d \in D} z_{j,t,d} = 1$ | $\forall j \in J$ |
| (6) $\sum_{j \in J} z_{j,t,d} \leq 1$ | $\forall t \in T, d \in D$ |
| (7) $e_{i,t,d} \geq e_{i,t-1,d}$ | $\forall i \in I, t \in T \setminus \{0\}, d \in D$ |
| (8) $l_{i,t,d} \leq l_{i,t-1,d}$ | $\forall i \in I, t \in T \setminus \{0\}, d \in D$ |
| (9) $l_{i,t,d} \geq \sum_{c \in C} x_{i,c,t,d}$ | $\forall i \in I, t \in T, d \in D$ |
| (10) $e_{i,t,d} \geq \sum_{c \in C} x_{i,c,t,d}$ | $\forall i \in I, t \in T, d \in D$ |
| (11) $v_{i,t,d} \geq e_{i,t,d} + l_{i,t,d} - \sum_{c \in C} x_{i,c,t,d} - 1$ | $\forall i \in I, t \in T, d \in D$ |
| (12) $\sum_{t \in T} x_{i,c,t,d} \leq 2$ | $\forall i \in I, c \in C, d \in D$ |
| (13) $o_{i,c,t,d} \geq x_{i,c,t,d} - x_{i,c,t-1,d}$ | $\forall i \in I, c \in C, t \in T \setminus \{0\}, d \in D$ |
| (14) $o_{i,c,t,d} \geq x_{i,c,t-1,d} - x_{i,c,t,d}$ | $\forall i \in I, c \in C, t \in T \setminus \{0\}, d \in D$ |
| (15) $\sum_{t=1}^{11} o_{i,c,t,d} \leq 2$ | $\forall i \in I, c \in C, d \in D$ |
| (16) $x_{i,c,t,d}, z_{j,t,d}, e_{i,t,d}, l_{i,t,d}, v_{i,t,d}, o_{i,c,t,d} \in \{0,1\}$ | $\forall i \in I, c \in C, t \in T, d \in D, j \in J$ |

TOELICHTING

- (1) Deze restrictie geeft aan dat iedereen in het Daltonuur vak 4, Dalton (zie de conversietabel in bijlage E), heeft. Dit is slechts een kleine aanpassing ten opzichte van de restrictie in het tussenurenmodel.
- (2) en (3) Deze restricties geven aan dat de fictieve lessen geen kleur mogen bevatten.
- (4) Deze restrictie is slechts een kleine aanpassing ten opzichte van de restrictie in het tussenurenmodel, namelijk het toevoegen van de index c.
- (5) t/m (8) Deze restricties hebben dezelfde vorm en dezelfde toelichting als bij het tussenurenmodel.
- (9) t/m (11) Deze restricties zijn slechts licht aangepast ten opzichte van de restricties van het tussenuren model. Het enige verschil is dat er nu niet gerekend wordt met $x_{i,t,d}$ om te bepalen of een klas les heeft, maar dat er gesommeerd wordt over alle vakken die de klas kan hebben.
- (12) Elke klas mag maximaal twee uur op een dag hetzelfde vak hebben, namelijk of één keer twee uur of één keer één uur.

- (13) Deze restrictie zorgt ervoor dat de variabele die de overgangen tussen wel en niet les aan geeft de waarde één krijgt als de klas in een lesuur les heeft en het lesuur daarvoor niet.
- (14) Deze restrictie zorgt ervoor dat de variabele die de overgangen tussen wel en niet les aangeeft de waarde één krijgt als de klas in een lesuur geen les heeft en het lesuur daarvoor wel.
- (15) Het is wenselijk dat een vak maar één keer per dag aangeboden wordt. Dit mag een blokkur zijn of een enkel uur. Dit is equivalent aan hooguit 2 overgangen van wel naar niet les of andersom. Het is hierbij nodig om een fictief lesuur aan het begin van de dag en aan het eind van de dag toe te voegen. Dit zorgt ervoor dat een klas niet het eerste en het tiende lesuur les krijgt in hetzelfde vak.
- (16) Alle variabelen zijn binair.

De combinatie van (13) en (14) sluit niet uit dat de variabele $o_{i,c,t,d}$ de waarde 1 toegekend krijgt als er eerst geen les was en nu ook niet (of eerst wel en nu ook). Dit is niet zorgwekkend. De resultaten van deze variabele kunnen dan afwijken van de definitie, maar voor het berekenen is het irrelevant. Mocht het nodig zijn de variabele echt 0 te hebben als er geen overgang is, omdat er bijvoorbeeld al twee overgangen op een dag zijn, dan is dit mogelijk. Het zal dus niet de oplossing beïnvloeden.

Ik ben mij ervan bewust dat dit model zoveel restricties heeft dat het zeer goed mogelijk is dat er geen toegelaten oplossingen gevonden kunnen worden. Indien dit het geval is zal er toch getracht worden een soort basisrooster te genereren. Dit zorgt ervoor dat geanalyseerd kan worden waarom er geen toegelaten oplossing bestaat voor bovenstaand model. Hierbij zullen restricties (12) en (15) als zwakke restricties behandeld worden. Dit houdt in dat hier indien mogelijk aan voldaan moet worden, maar dat als er geen toegelaten oplossing is deze geschonden mogen worden. Hier komt het strafpunten systeem zoals in het voorwoord genoemd een klein beetje in terug. Zo zal er een waarde toegekend worden aan het overtreden van deze restricties en deze zal opgeteld worden bij de doelfunctiewaarde. Door het minimaliseren van de doelfunctie zal er getracht worden geen strafpunten te verkrijgen en dus aan de restricties te voldoen.

Ik zal de overtreding van restrictie (15), dat alle lessen van een vak aaneengesloten moeten zijn, minder streng bestraffen als restrictie (12). Dit omdat ik meen dat het minder erg is om over de dag verspreid twee losse uren van een vak te hebben dan drie aaneengesloten uren. Dit idee komt ook voort uit de opvatting in het onderwijs dat vakken over de week verspreid zouden moeten worden.

5. Resultaten

5.1. Datatransformatie

De data zijn aangeleverd als volwaardige huidige roosters voor de klassen en zijn terug te vinden op de website van het Stedelijk Dalton Lyceum¹. Om deze data voor het onderzoek te kunnen gebruiken zijn deze getransformeerd naar een matrix waarin de rijen de klassen voorstellen en de kolommen de docenten. De waarde in de matrix geeft aan hoe vaak deze klas en docent elkaar in één week moeten treffen. Voor deze transformatie is gebruikt gemaakt van conversietabellen die terug te vinden zijn in bijlage A en bijlage B. Het grondrooster dat nu ontstaat is terug te vinden in bijlage C.

¹ <http://dalton-dordrecht.nl/overkampweg/roosters/klassenroosters/klassen.htm>

5.2.Data-analyse

De data behoeften geen grondige analyse op fouten, omdat dit het rooster is dat in de praktijk gebruikt wordt. Het aantal lessen per vak is door de school nagelopen alvorens het rooster in gebruik te nemen. Wel viel op dat bijvoorbeeld klas hv2b twee lessen meer heeft per week dan klas hv2a. Dit lijkt onwaarschijnlijk, maar is terug te voeren op het gegeven dat een aantal vakken slechts een halfjaar per klas gedoceerd worden. Zo zijn er nog een paar van deze situaties die allemaal naar dit gegeven terug leiden.

5.3.Fase 1

5.3.1. Heuristiek

De eerder beschreven heuristiek is geïmplementeerd in MatLab, versie 7.0.0.19920(R14) en toegepast op het grondrooster zoals terug te vinden in bijlage C. De resultaten hiervan zijn terug te vinden in bijlage D. In deze tabel staan de rijen voor het klasnummer (zoals ontstaan via conversietabel A) en de kolommen staan voor het kleurnummer. Het getal in het betreffende element betekent het docentnummer van de docent die de klas les geeft in die kleur. Indien er een nul in het element staat heeft de klas in die kleur geen enkele les.

In het hoofdstuk methodologie was reeds opgemerkt dat het aantal kleuren minimaal gelijk is aan de hoogste graad van alle knopen. In dit probleem is die hoogste graad gelijk aan 40. (Klas tg3 heeft 40 lessen die geroosterd moeten worden). Zoals te zien is in de tabel in bijlage D zijn slechts 40 kleuren gebruikt. Verder valt op te merken dat Matlab deze heuristiek in 0.062 seconde uitvoert.

Vanwege het gegeven dat er gebruik gemaakt is van random keuzes gedurende de heuristiek heb ik de heuristiek 10.000 keer gerund. Dit was om te evalueren hoeveel keer deze random keuze voorkomt. Als dit namelijk vaak voorkomt kan het dus lucratief zijn om deze selectie niet random uit de uniforme verdeling te bepalen, maar op een andere manier. Dit is uiteraard alleen aan de orde als blijkt dat de heuristiek niet de gewenste uitkomst geeft. Bij deze evaluatie run bleek dat er 10.000 keer 40 kleuren als resultaat gegeven is. Dit betekent nadrukkelijk niet dat het voor elk grondrooster het minimum aantal kleuren zal genereren; het is slechts een constatering dat voor dit specifieke grondrooster het minimum aantal kleuren gebruikt wordt door de heuristiek.

In deze 10.000 keer zijn ook nog een aantal andere statistieken verzameld die hieronder opgesomd worden:

- Voor het bepalen van de moeilijkste klas wordt er elke keer bij het genereren van 1 rooster 804 keer gebruikt gemaakt van het random kiezen van een klas.
- Voor het bepalen van de moeilijkste docent bij de klas wordt er bij het genereren van 1 rooster gemiddeld 647 keer gebruik gemaakt van het random kiezen van een docent.
- Het maximum aantal keer dat de moeilijkste docent bepaald werd door middel van random selectie was 673 keer.
- Het minimum aantal keer dat de moeilijkste docent bepaald werd door middel van random selectie was 619 keer.

5.3.2. Maximum koppelingen

Vanwege de gewenste uitkomsten van de heuristiek is deze methode niet geïmplementeerd.

5.4.Fase 2

5.4.1. Heuristiek

Het initiële model zal met een exact algoritme opgelost worden in AIMMS 3.8 met de solver XA14. Vervolgens zal met Matlab, versie 7.0.0.19920(R14), getracht worden, indien noodzakelijk, deze oplossing te verbeteren.

Voor de implementatie in AIMMS is gebruik gemaakt van element parameters en subsets voor alle restricties die niet voor de hele index gelden. Indien het één element uit een index betrof was het mogelijk een subset te gebruiken, maar er is gekozen voor een element parameter. Verder is er bij de implementatie voor gekozen om de restricties die t met $t - 1$ vergelijken allemaal individueel in te voeren. Dit omdat AIMMS problemen krijgt met de uitdrukken $t - 1$ als je data niet perfect op volgorde staan. Uiteraard zijn ze op volgorde ingevoerd, maar als ik een element parameter aanmaak wordt dat element in die index vooraan geplaatst. Dan is de volgorde weg en ontstaan er problemen met deze restricties. Beide opmerkingen gelden ook voor subparagraaf 5.4.2.

Voor deze heuristiek is er voor gekozen om restrictie (7), van het plaatsen van de Daltonuren in het vijfde lesuur, los te implementeren. Deze is opgesplitst in vijf restricties, waarbij vooraf bepaald is dat kleur 41 op maandag het vijfde uur geplaatst moet worden, kleur 42 op dinsdag enzovoort. Dit is gedaan om het model sneller te laten runnen. Alle wisselingen binnen het Daltonuur zorgen namelijk voor dezelfde oplossing en moeten dus allemaal geëvalueerd worden. Door de losse restricties hoeft dit niet meer. De oplossing blijft hetzelfde, deze is alleen sneller gevonden.

Het totale aantal betrokken partijen bedraagt 964, waarbij de Daltonuren meegeteld zijn. Dit betekent dat $0 \leq r \leq 192$. Via de 'trial and error' methode is ontdekt dat voor case 2, 3 en 4 zelfs geldt dat $0 \leq r \leq 168$. Bij waarden tussen 168 en 192 werd direct de melding gegeven dat er geen toegelaten oplossing gevonden kon worden. Dit was te verwachten, omdat er nog maar 7 lesuren op één (of twee) dagen over zijn. In een lesuur kunnen maximaal 24 klassen les krijgen dus er zijn maximaal $7 \cdot 24 (= 168)$ betrokken partijen op die dag.

Ter illustratie van de uitkomsten is hieronder de plaatsing van case 1, waarbij alles vrij geroosterd mag worden, weergegeven. Op basis van deze uitkomst kunnen roosters gegenereerd worden. Hiervoor dient er per klas bekeken te worden in welke kleuren er een les zat en dan wordt dit uur nu geroosterd op de plek waar de kleur geplaatst is. De plaatsingen van case 2, 3 en 4 zijn terug te vinden in respectievelijk bijlage F, G en H.

In de plaatsing staat de rij voor het lesuur, de kolom voor de dag en het getal voor de kleur die geplaatst is. De laatste vijf kleuren zijn in de roosters vermeld als Daltonuur, zodat de plaatsingen makkelijker te vergelijken zijn met de uitkomst van het exacte algoritme.

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	32	36	-	6	1
2	24	35	-	3	12
3	16	23	-	7	21
4	14	20	8	28	19
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	31	11	17	18	5
7	26	9	13	15	33
8	25	29	2	4	37
9	30	39	10	34	38
10	27	-	22	40	-

Bij de plaatsing dient opgemerkt te worden dat hieruit heel snel andere plaatsingen gevormd kunnen worden. Er zijn een aantal clusters van kleuren waarin dezelfde klassen les hebben:

- Cluster A: de kleuren 1 tot en met 32
- Cluster B: de kleuren 33 en 34
- Cluster C: de kleuren 37 en 38

De kleuren binnen de clusters mogen onderling gewisseld worden, zonder dat er tussenuren ontstaan. Dit kan bijvoorbeeld beter uitkomen met de vakken die in die kleur gegeven worden. Sommige kleuren zitten niet in een cluster, omdat ze uniek zijn. Wellicht dat deze ook gewisseld kunnen worden zonder gevolgen, maar dat is niet zo triviaal.

In onderstaande tabel zijn statistieken verzameld van de vier verschillende cases.

	Case 1	Case 2	Case 3	Case4
Gebruikte waarde voor r	150	132	150	150
Aantal iteraties	1134	553	271	540
Beste LP grens	240	240	240	240
Gevonden oplossing	240	240	240	240
Verskil tussen beste LP grens en gevonden oplossing	0%	0%	0%	0%
Rekentijd (in seconde)	0.61	0.31	0.19	0.30
Totaal gebruikte tijd (in seconde)	0.64	0.31	0.26	0.26

De gevonden oplossingen met een waarde van 240 zijn optimaal. Dit is als volgt te beredeneren: in het Daltonuur zijn er 24 betrokken partijen. Aan het einde en aan het begin van de dag moet dit altijd 0 worden. Dit gebeurt dus aan het begin van de dag en aan het einde van de dag, maandag tot en met vrijdag. Dus 10 keer van 24 naar 0 betekent minimaal 240.

Bij de vierde case dient een aparte opmerking geplaatst te worden: hoewel de heuristiek uit de eerste fase het minimale aantal kleuren gevonden heeft is het onmogelijk om deze case te roosteren, omdat er één uur te weinig in een week zit. Met slechts een kleine aanpassing is er echter wel een rooster te creëren. Deze aanpassing houdt in dat op het achtste uur op dinsdag de kleur geplaatst wordt met het minste aantal klassen. Hierdoor zijn er zo min mogelijk docenten afwezig bij het eerste uur van de vergadering.

Bij het vergelijken van case 2 en 3 valt op dat er meerdere optimale oplossingen voor dit probleem zijn. Uiteraard zou het voor case 3 ook mogelijk zijn om het rooster van case 2 te gebruiken en in zijn geheel de dinsdag en de donderdag te verwisselen. Dit is hier niet gebeurd, ook niet als je kleuren binnen de clusters zou wisselen, dus er zijn meerdere roosters zonder tussenuren met de laatste drie uur op een dag in de week vrij.

Vervolgens zijn op basis van bovenstaande tabel roosters gegenereerd en hieruit bleek dat er nul tussenuren waren voor de klassen! Dit is geen garantie dat het altijd zo werkt, maar dat het nu zo mooi uitkomt komt zeer waarschijnlijk door de mooie uitkomst van fase 1. In fase 1 is het zo dat als je de klas langsloopt er geen kleuren zijn die leeg zijn (tot de laatste kleur). Hierdoor is het zo dat als een klas meer betrokken partijen heeft het altijd de klassen bevat die in de kleur zitten met minder betrokken partijen.

5.4.2. Exact algoritme

Bij de implementatie van beide modellen in AIMMS 3.8 zijn een aantal aanpassingen gedaan om sneller tot de oplossing te komen:

Ten eerste is er voor gekozen om voor elke klas en op elke dag te definiëren dat de variabele die aangeeft of de klas op of voor het eerste lesuur les heeft gelijk moet zijn aan de variabele die aangeeft of de klas op het eerste lesuur les heeft. Het is toegestaan om dit te definiëren zonder de oplossing te veranderen, omdat er geen lesuren voor het eerste lesuur zijn en dit dus altijd waar is.

Ten tweede is er voor gekozen om iets soortgelijks te doen voor de variabele die aangeeft of een klas op of na het laatste lesuur les heeft. Dit moet met een soortgelijke redenatie namelijk ook altijd gelijk zijn aan de variabele die aangeeft of de klas op het laatste lesuur les heeft.

Ten derde heb ik alle variabelen die aangeven of een klas op of voor een lesuur na het Daltonuur les heeft gelijk gesteld aan 1. Dus deze variabelen krijgen voor lesuur 6 tot en met 10 de waarde 1 voor alle klassen op alle dagen. Dit verandert de oplossing niet, omdat elke klas in het Daltonuur les heeft en in alle volgende lesuren dus minimaal een les voor dat lesuur gehad heeft.

Ten slotte is er iets soortgelijks gedaan voor de variabelen die aangeven of een klas op of na een lesuur voor het Daltonuur les heeft. Ook dit is altijd waar, omdat elke klas in het Daltonuur les heeft en in alle vorige lesuren dus minimaal een les na dat lesuur gaat krijgen.

Model voor een minimaal aantal tussenuren zonder verdere beperkingen

Ter illustratie van de uitkomsten is hieronder weer de plaatsing van case 1, waarbij alles vrij gerosterd mag worden, weergegeven. De plaatsingen van case 2, 3 en 4 zijn terug te vinden in respectievelijk bijlage I, J en K. Hierbij gelden dezelfde instructies en clusters als in de vorige subparagraaf.

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	27	-	15	28	33
2	30	-	29	13	10
3	23	37	22	7	18
4	19	26	9	6	5
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	17	32	8	11	12
7	24	34	20	14	31
8	21	38	4	25	2
9	16	39	3	-	35
10	1	40	-	-	36

Bij de vergelijking van deze uitkomst met de uitkomst van de heuristiek valt op dat er een ander rooster gegenereerd is, zelf wanneer het wisselen van dagen of binnen clusters als hetzelfde rooster gerekend wordt. Dit betekent dat er meerdere optimale oplossingen zijn voor dit probleem.

In onderstaande tabel zijn statistieken verzameld van de vier verschillende cases.

	Case 1	Case 2	Case 3	Case4
Aantal iteraties	3444	3581	2965	2617
Beste LP grens	0	0	0	0
Gevonden oplossing	0	0	0	0
Vershil tussen beste LP grens en gevonden oplossing	0%	0%	0%	0%
Rekentijd (in seconde)	7,23	6,91	5,91	4,63
Totaal gebruikte tijd (in seconde)	7,41	6,97	6,17	4,86

Ook hier valt op bij de vergelijking van de uitkomst van case 2 met case 3 op dat er meerder roosters zijn waarbij de laatste drie uur op een dag vrij blijven, zelfs als alle wisselingen van dagen en binnen clusters als dezelfde roosters worden gezien.

Model voor een minimaal aantal tussenuren met bepaalde voorwaarde

Voor de data van dit model moesten een aantal tabellen aan elkaar gekoppeld worden. Hiervoor werd er per klas bekeken in welke kleur ze welke docent hebben. Vervolgens is bekeken welke vakken deze docent aan deze klas geeft. En tenslotte is bepaald welk vak de klas in deze kleur kreeg. Hier zijn twee situaties te onderscheiden:

- I. De docent geeft maar één vak aan de klas. Dan heeft de klas dus dat vak in deze kleur.
- II. De docent geeft meerdere vakken aan de klas. Dan heeft de klas een van die vakken in deze kleur. Welk vak dit werd is willekeurig bepaald. Hierbij is er uiteraard wel voor gezorgd dat uiteindelijk het aantal lessen per vak klopt.

Bij het runnen van dit model in AIMMS gaf de solver niet direct een melding dat er geen toegelaten oplossing is. Dit was hoopvol, maar na ongeveer een half uur runnen en 15.200 iteraties was er nog geen toegestane oplossing gevonden.

Hierop is besloten het model de ruimte te geven om de restricties (12) en (15) te schenden. Met verschillende strafwaardes in de verhouding 10:2, respectievelijk, is getracht een basis oplossing te genereren. Ook hier werd niet direct een melding gegeven dat er geen toegelaten oplossing was, maar werd er niet binnen afzienbare tijd een toegelaten oplossing gegeven. Uiteraard zijn er toegelaten oplossingen voor deze situatie, omdat alle oplossingen van het vorige model nu ook toegestaan zijn. Ze kunnen echter heel slecht zijn, omdat nu ook van belang is welke vakken er gegeven worden.

6. Conclusie

In deze scriptie is een rooster gegenereerd voor een middelbare school. Hierbij is begonnen met een lijnkleuringsprobleem welke met een heuristiek opgelost is. Deze heuristiek resulteerde in het minimum aantal kleuren voor het specifieke grondrooster dat gebruikt is voor deze scriptie. Vervolgens is op verschillende manieren een rooster getracht te maken door plaatsing van deze kleuren in een week. Hierbij zijn ook verschillende cases aan bod gekomen.

Één daarvan was de eis dat de dinsdagmiddag en donderdagmiddag geen lessen mochten bevatten in verband met vergaderingen en CSM activiteiten. Dit is onmogelijk te roosteren, omdat er dan simpelweg één lesuur te weinig in een week zit. Mijn advies hierin is om de kleur met de minste klassen op dinsdag het achtste uur te plaatsen, zodat zo min mogelijk docenten het eerste uur van de vergadering missen.

Voor het model met betrokken partijen is het minimum aantal overgangen gerealiseerd. De uitkomst van deze kleurplaatsingen heeft zelfs geresulteerd in een rooster zonder tussenuren voor de klassen. Deze mooie uitkomst is waarschijnlijk ingegeven door de uitkomst van fase 1 dat klassen altijd in opeenvolgende kleuren les hebben.

Aangezien de heuristiek tot de optimale oplossing heeft geleid, heeft het exacte algoritme zonder voorwaarde uiteraard ook de optimale oplossing bereikt. Het exacte algoritme met voorwaarde heeft helaas geen resultaat gegenereerd. Het gegeven dat het model zonder voorwaarde zeer snel een oplossing vindt en het model met voorwaarde niet, resulteert mijns inziens in het volgende: het model met voorwaarde, met zwakke restricties (12) en (15), moet een oplossing hebben, echter het model heeft te veel variabelen om binnen afzienbare tijd een oplossing te genereren. Het model zou dus nog een keer op een sneller computer gerund moeten worden.

Alles samenvattend is het in deze scriptie gelukt een rooster te genereren zonder tussenuren voor de klassen, waarbij geen rekening wordt gehouden met hoe vaak een bepaald vak op een dag gegeven wordt. Ook wordt er niet gekeken of deze uren aaneengesloten zijn.

7. Literatuurlijst

Bondy, J.A. en Murty, U.S.R. (1976). Graph theory with applications, Macmillan, London.

Mulder, H.M. (2005). Wiskundige Methoden, Syllabus Econometrisch Instituut EUR.

Werra, D. de (1985). 'An Introduction to Timetabling.' European Journal of Operational Research, 19, 151-162

8. Bijlagen

Bijlage A: Conversietabel klassen

Nummer	Klas
1	b1a
2	b1b
3	b1c
4	b1e
5	b1f
6	b1g
7	b1k
8	b1m
9	b1p
10	hv2a
11	hv2b
12	a2a
13	a2b
14	g2a
15	g2b
16	ta2
17	tg2
18	h3a
19	a3a
20	a3b
21	a3c
22	g3a
23	ta3
24	tg3

Bijlage B: Conversietabel docenten

Nummer	Afkorting		Nummer	Afkorting
1	AlEm		41	LeCo
2	BaFr		42	MiJa
3	BaRa		43	MuRi
4	BeJa		44	NuMa
5	BeMa		45	PEle
6	BeNe		46	PaRo
7	BiLe		47	RiCl
8	BiWi		48	ScWi
9	BoEl		49	SeLi
10	BoHe		50	StDi
11	BrCo		51	StLa
12	BrKa		52	StRo
13	BrNc		53	StSj
14	BruA		54	StWi
15	BruN		55	SvAn
16	CiYi		56	TiPa
17	DiEm		57	TrDa
18	DuJa		58	VeWi
19	EeRi		59	ViMa
20	EiHa		60	VIAn
21	GiKo		61	VoMa
22	GoRo		62	VosM
23	GrCe		63	VuHa
24	GrMa		64	WaaJ
25	GuJo		65	WiCe
26	HaLi		66	WiWi
27	HatM			
28	HeTo			
29	HoEl			
30	HoNe			
31	HoOt			
32	HorM			
33	HuAl			
34	IrKe			
35	KIFr			
36	KlKe			
37	KoJa			
38	KoTh			
39	KuiJ			
40	KuJo			

Bijlage C: Grondrooster

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
1	0	0	0	3	4	0	0	0	0	0	2	0	0	0	0	0	0	0	0	2	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	2	0	0	
3	0	4	0	0	0	3	0	0	0	0	2	3	0	0	4	0	0	0	0	2	0	0	
4	0	0	0	5	0	0	0	0	0	0	0	0	0	0	2	0	0	2	0	0	0	0	
5	4	0	0	0	0	0	0	0	0	0	2	0	1	0	2	0	0	2	0	0	0	3	0
6	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	3	0
7	0	0	0	0	0	3	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	4	0	0	3	0	0	1	0	0	0	0	0	0	0	0	0	0
9	0	0	0	3	0	0	4	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	4	0	0	3	3	0	0	0	0	0	0	0	0	0	3	0	0	2	0	0	
11	0	3	0	0	0	0	3	0	0	0	2	0	0	0	0	0	0	0	0	2	3	0	
12	0	0	0	0	0	3	0	0	0	0	2	4	0	0	2	0	0	0	3	2	0	0	
13	3	0	0	0	0	0	0	0	0	3	0	0	0	0	2	0	3	0	0	2	0	0	
14	0	0	0	0	0	0	0	0	0	0	2	0	0	4	0	4	3	0	0	2	0	0	
15	0	3	0	0	0	0	0	0	0	0	3	0	0	0	2	0	0	0	0	2	0	0	
16	0	0	0	3	0	0	3	3	0	0	2	0	0	0	0	0	0	2	0	0	0	0	
17	0	0	0	0	0	2	3	0	0	0	0	0	5	0	0	0	0	2	0	0	0	0	
18	0	5	0	2	0	0	0	0	2	0	0	0	0	2	3	3	0	0	0	0	0	0	
19	0	4	0	0	0	3	0	0	0	0	0	0	0	3	0	2	0	0	2	0	0	0	
20	0	0	0	0	0	0	0	0	2	0	0	0	0	0	3	0	2	0	0	0	0	0	
21	0	0	0	0	0	2	0	0	6	0	0	0	0	2	0	0	2	0	0	0	0	0	
22	0	0	0	2	0	0	0	0	0	0	0	2	3	0	2	2	0	0	0	2	0	0	
23	0	0	0	2	0	0	0	4	0	0	0	0	0	2	0	0	0	0	0	0	0	2	
24	0	0	0	0	0	0	0	0	0	2	0	0	6	0	0	0	0	0	0	0	0	2	

	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43
1	0	0	0	0	0	0	0	0	0	0	0	0	3	0	2	0	0	0	0	0	0
2	2	0	0	0	3	0	0	0	0	4	0	0	3	2	0	0	0	0	0	0	0
3	0	3	0	0	0	0	0	2	0	0	0	0	0	0	0	4	0	0	0	0	0
4	0	3	0	0	0	0	1	0	0	4	0	0	0	0	0	4	0	0	0	0	0
5	0	3	0	0	5	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	3
6	0	3	0	0	3	0	0	0	0	4	0	0	0	2	0	0	0	0	0	0	0
7	2	0	0	7	0	0	1	0	0	0	0	0	3	0	2	0	0	0	0	2	0
8	2	0	0	0	0	0	0	0	0	0	0	0	3	2	2	0	0	0	0	2	4
9	2	0	0	0	0	0	0	0	0	0	0	0	3	2	2	0	0	0	0	4	0
10	2	0	2	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
11	0	2	2	0	3	0	0	0	0	0	0	4	0	0	0	0	0	0	0	1	0
12	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0
13	0	0	0	0	0	0	0	2	0	0	0	5	0	0	0	0	0	0	0	0	0
14	0	0	0	0	2	0	2	0	2	3	0	0	2	0	0	0	0	0	0	0	0
15	2	0	0	0	2	0	2	0	2	0	0	4	0	0	0	0	0	0	0	1	0
16	0	0	0	5	0	0	0	0	2	0	0	4	2	0	0	0	0	0	0	1	0
17	2	0	0	0	0	0	0	0	2	0	4	0	0	2	0	0	0	0	0	1	0
18	2	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	3	2	2	0
19	0	0	0	0	0	0	0	2	0	0	3	0	0	0	0	2	0	3	0	0	0
20	0	0	0	0	2	0	0	2	0	4	0	3	0	0	0	0	0	3	0	0	0
21	2	0	0	0	0	0	0	0	0	4	3	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	2	3	0	0	4
23	3	0	0	4	0	0	0	0	2	0	3	0	0	3	2	0	0	0	2	2	0
24	2	0	0	0	0	2	0	0	2	0	0	2	0	2	2	0	0	0	3	2	0

	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66
1	0	0	0	0	0	0	0	6	0	1	0	2	0	4	0	0	0	0	2	3	0	2
2	0	4	0	0	0	0	2	0	0	1	0	2	6	0	0	0	0	0	0	0	0	2
3	0	4	0	0	0	0	0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	2
4	0	0	3	0	0	0	2	0	0	1	0	2	0	0	0	0	0	0	2	3	0	2
5	0	0	0	0	0	0	0	4	0	1	2	2	0	0	0	0	0	0	0	0	0	0
6	1	0	3	0	0	0	2	0	0	1	0	2	0	0	0	0	0	0	2	0	0	4
7	0	0	0	0	4	0	0	0	0	1	2	0	0	4	0	0	0	0	0	0	0	2
8	0	0	0	0	4	0	2	0	0	1	0	0	0	0	0	7	0	0	0	0	0	2
9	1	0	0	0	4	0	0	0	0	1	0	0	0	0	0	5	0	0	0	0	0	2
10	0	0	0	0	0	0	2	0	2	0	0	1	0	0	0	0	0	4	0	0	0	0
11	0	0	0	0	0	0	2	4	3	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	3	0	0	0	0	0	0	2	0	0	1	0	0	0	0	0	4	0	0	0	0
13	0	3	0	0	0	0	2	4	2	0	0	1	0	0	0	0	0	0	0	0	0	0
14	2	0	0	0	0	0	0	0	0	0	2	1	0	4	0	0	0	0	0	0	0	0
15	2	0	0	0	0	0	2	0	0	0	0	0	4	0	0	0	0	0	0	3	0	0
16	0	0	0	0	5	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	4	0	2	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	2	0	0	0	0	0	2	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	2	0	0	2	0	0	2	0	2	0	0	0	0	0
20	0	0	0	0	0	0	2	2	0	0	0	3	0	0	2	0	4	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	2	0	3	2	0	4	0	0	0	0	0
22	0	0	0	2	0	2	0	2	0	0	0	2	0	0	0	0	2	0	0	0	2	0
23	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	3	0	2	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0

Bijlage D: Uitkomst fase 1 algoritme

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	20	52	56	4	66	52	11	56	52	20	35	4	52	4	37	52	37	64	35	58
2	36	35	32	27	27	32	56	20	32	36	66	23	20	32	12	57	35	66	12	46
3	15	66	20	56	11	56	6	2	66	38	38	6	11	46	6	30	2	15	46	12
4	32	51	66	15	15	38	51	24	38	32	64	32	56	38	56	4	4	38	54	32
5	52	56	52	24	52	27	27	52	56	27	43	15	18	24	27	24	43	43	18	21
6	27	32	27	66	32	24	66	36	51	24	24	56	66	66	51	32	32	18	3	3
7	11	49	11	6	23	66	26	49	23	44	6	42	37	26	49	35	44	58	42	26
8	51	60	23	35	49	51	35	35	7	7	13	7	49	23	36	60	7	54	36	49
9	49	4	7	60	4	11	4	23	49	35	11	44	42	49	60	44	49	42	23	37
10	53	7	51	62	6	7	20	51	17	6	32	17	32	6	32	23	56	23	53	62
11	21	20	24	7	7	2	34	27	34	2	20	2	21	34	34	42	53	11	11	7
12	24	11	6	46	53	53	12	12	20	15	46	11	24	56	62	62	15	20	6	38
13	30	46	46	20	17	17	17	34	1	34	52	46	53	52	10	20	34	52	56	1
14	35	27	58	32	20	58	32	11	27	58	58	31	55	11	14	56	20	17	29	16
15	2	2	42	2	51	15	31	31	11	29	34	57	15	51	57	27	57	34	27	64
16	34	34	35	26	34	34	49	26	35	11	7	26	26	18	11	7	31	4	8	55
17	7	6	49	23	60	23	7	44	60	49	33	51	6	60	7	49	60	44	49	51
18	42	9	2	40	42	4	59	40	2	23	4	14	2	15	15	15	26	9	41	14
19	17	38	53	17	2	6	2	15	53	33	56	40	38	61	59	40	30	59	15	15
20	40	17	30	51	9	61	15	32	15	51	15	52	34	27	52	17	9	27	34	61
21	6	58	33	61	61	33	23	61	58	56	23	61	33	9	58	14	17	32	32	56
22	56	15	15	52	56	20	52	4	4	12	40	13	13	12	20	43	16	29	16	13
23	46	42	26	42	26	26	46	46	26	37	14	33	14	36	4	33	41	37	4	23
24	23	23	34	34	46	46	60	60	46	13	60	36	60	13	44	13	42	36	13	31

	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	64	52	54	58	11	63	63	5	35	58	64	5	66	5	5	58	0	0	0	0
2	35	46	51	57	57	56	54	57	46	27	46	23	57	51	12	57	0	0	0	0
3	30	2	38	24	12	2	38	12	15	54	20	46	24	15	24	46	0	0	0	0
4	29	18	24	66	47	47	18	4	24	63	4	4	64	63	64	47	0	0	0	0
5	21	54	1	15	27	55	13	30	1	11	21	30	1	11	1	55	0	0	0	0
6	56	47	63	21	3	18	36	27	47	3	63	21	47	21	54	45	0	0	0	0
7	58	55	66	6	35	26	35	26	49	26	58	55	44	54	58	26	37	44	29	0
8	60	10	43	60	60	49	37	42	37	10	66	66	43	60	10	42	43	43	60	0
9	54	36	45	44	66	37	7	60	44	35	60	60	42	35	7	36	42	7	66	0
10	3	20	3	7	25	17	32	25	62	62	3	3	0	0	0	0	0	0	0	0
11	53	24	52	51	21	52	52	51	25	52	27	53	25	27	0	0	0	0	0	0
12	46	62	6	19	62	12	12	38	38	19	19	38	0	0	0	0	0	0	0	0
13	51	15	53	1	30	34	10	15	34	51	10	52	0	0	0	0	0	0	0	0
14	16	29	32	45	45	16	17	16	14	14	55	17	35	31	14	0	0	0	0	0
15	20	64	34	23	23	64	11	45	57	45	29	11	20	34	0	0	0	0	0	0
16	4	49	55	31	8	42	8	49	7	49	26	18	4	49	0	0	0	0	0	0
17	42	31	33	13	44	13	31	18	13	36	18	13	36	33	33	13	0	0	0	0
18	23	16	2	26	2	53	59	53	16	40	16	41	0	0	0	0	0	0	0	0
19	33	30	40	20	20	33	6	2	56	6	61	2	0	0	0	0	0	0	0	0
20	32	32	59	34	32	61	56	56	40	61	30	59	40	56	0	0	0	0	0	0
21	14	9	9	59	9	59	9	6	32	17	32	9	0	0	0	0	0	0	0	0
22	40	43	61	29	43	40	61	29	43	29	48	48	39	65	50	50	39	65	0	0
23	36	8	8	36	33	22	23	31	8	41	31	8	23	22	0	0	0	0	0	0
24	31	13	44	42	13	10	44	41	10	48	41	22	41	37	44	37	22	28	28	48

Bijlage E: Conversietabel vakken

Nummer	Afkorting	Volledige naam
1	ak	Aardrijkskunde
2	bi	Biologie
3	bv	Beeldende vorming
4	dalton	Daltonuur
5	dr	Drama
6	du	Duits
7	dv	Denkvaardigheden
8	ec	Economie
9	en	Engels
10	fa	Frans
11	fi	Filosofie
12	gr	Grieks
13	gs	Geschiedenis
14	la	Latijn
15	lo	Lichamelijke oefening
16	mentor	Mentoruur
17	mu	Muziek
18	na	Natuurkunde
19	ne	Nederlands
20	nt	Natuur en techniek
21	sk	Scheikunde
22	sl	Studielessen
23	wi	Wiskunde

Bijlage F: Heuristiek, de plaatsing van case 2

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	36	30	24	37	40
2	4	30	3	11	39
3	8	9	5	23	29
4	18	26	1	16	6
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	15	20	17	34	13
7	12	27	14	33	2
8	25	22	38	CSM	21
9	10	28	-	CSM	7
10	32	19	-	CSM	35

Bijlage G: Heuristiek, de plaatsing van case 3

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	39	34	5	38	19
2	37	31	28	33	29
3	36	24	25	1	3
4	35	23	22	20	4
5	Daltonuur	Daltonuur	Daltonuur	Daltonuur	Daltonuur
6	10	8	11	30	18
7	6	14	15	26	2
8	9	Vergadering	27	32	21
9	13	Vergadering	-	17	7
10	40	Vergadering	-	16	12

Bijlage H: Heuristiek, de plaatsing van case 4

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	6	30	2	26	28
2	23	4	17	15	25
3	5	3	29	31	13
4	20	32	12	11	27
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	14	24	33	16	1
7	21	22	35	7	8
8	11	40/Vergadering	36	CSM	10
9	18	Vergadering	38	CSM	19
10	34	Vergadering	39	CSM	37

Bijlage I: Exact zonder voorwaarde, de plaatsing van case 2

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	19	36	8	37	34
2	32	23	28	22	33
3	26	4	29	16	21
4	12	9	30	31	18
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	27	11	2	7	6
7	13	14	38	20	10
8	25	24	39	CSM	1
9	15	35	40	CSM	5
10	17	-	-	CSM	3

Bijlage J: Exact zonder voorwaarde, de plaatsing van case 3

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	33	9	11	31	17
2	18	23	12	25	32
3	10	4	14	24	20
4	2	8	3	19	7
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	27	34	29	13	5
7	6	36	28	1	30
8	16	Vergadering	37	21	22
9	35	Vergadering	38	40	15
10	-	Vergadering	39	-	26

Bijlage K: Exact zonder voorwaarde, de plaatsing van case 4

	maandag	dinsdag	woensdag	donderdag	vrijdag
1	16	29	25	10	38
2	18	2	3	8	34
3	32	19	28	4	17
4	31	20	5	13	15
5	Dalton	Dalton	Dalton	Dalton	Dalton
6	12	6	21	9	27
7	30	24	23	22	26
8	7	40/Vergadering	1	CSM	14
9	11	Vergadering	35	CSM	37
10	33	Vergadering	36	CSM	39