

Bacheloropdracht: Roosteren voor Scala

CINTHA KOOISTRA

Begeleider: Gerhard Post
Universiteit Twente

Enschede, 29 juni 2018

Samenvatting

In deze bacheloropdracht is een programma geschreven in C# dat een rooster maakt voor de culturele instelling Scala. Scala maakte tot nu toe altijd het rooster met de hand. Aangezien Scala steeds meer groeit, is dit haast niet meer te doen. Zodoende zou een programma dat voor hen een rooster maakt erg nuttig zijn. Naast zorgen dat het programma de gegevens van Scala kan inlezen, moest ervoor gezorgd worden dat er een model, bestaande uit hard en soft constraints, werd geïmplementeerd in het programma. De hard constraints waren nodig om een geldig rooster te krijgen en de soft constraints weerspiegelden de voorkeuren. Samen zorgde de constraints ervoor dat het programma een rooster maakte die voldeed aan de punten die Scala graag in het rooster terug zou willen zien. Helaas zijn in dit korte tijdsbestek niet alle constraints in het model verwerkt. Door ontbrekende gegevens en het feit dat er onvoldoende geheugen beschikbaar was op de computer waarop het programma draaide, was het niet mogelijk om een rooster te produceren dat representatief is voor volgend schooljaar en daarmee te onderzoeken of het roosterprobleem op een exacte manier kan worden opgelost door gebruik te maken van het gemaakte programma. Echter, van alle constraints die in het programma zijn geïmplementeerd, is getest dat ze naar behoren werken. Doordat nog niet alle voorwaarden van Scala in het programma verwerkt zijn en het programma nog geen grote datasets aankan, is het programma nu nog niet klaar om gebruikt te worden door Scala.

Sleutelwoorden — Roosterprobleem, Integer linear programming (ILP), C#, CPLEX Optimizer

INHOUDSOPGAVE

1	Inleiding	3
2	Literatuuronderzoek	3
3	Probleemoriëntatie	4
3.1	Constraints	5
3.1.1	Hard constraints	5
3.1.2	Soft constraints	6
3.1.3	Toelichting gedeeltelijk meegenomen constraints	6
3.2	Aannames	7
4	Model	8
4.1	Verzamelingen	8
4.2	Variabelen	9
4.3	Doelfunctie	9
4.4	Constraints	10
5	Resultaten	11
5.1	Toelichting programma	11
5.2	Constraints controleren	14
6	Discussie en aanbevelingen	21
7	Conclusie	23
8	Dankwoord	23
9	Literatuurlijst	24
10	Appendix	25
10.1	Punten waar tegenaan werd gelopen	25
10.2	Layout data Scala	26

1. INLEIDING

Scala Centrum voor de Kunsten is een culturele instelling die veel workshops en lessenseries geeft op scholen in de gemeentes Meppel, De Wolden, Hoogeveen, Staphorst, Westerveld, Weststellingwerf en Steenwijkerland. Scala is werkzaam in deze gemeentes op ruim 100 scholen. Scala biedt workshops aan in alle klassen in de kunstdisciplines muziek, theater, dans, media, beeldend en erfgoed. De scholen kunnen zelf kiezen welke workshops ze willen, waarna Scala de juiste docent en tijdstip zoekt om hiervoor naar de school te komen. Tot nu toe werd deze planning met de hand gedaan, maar doordat Scala steeds meer groeit, is dit bijna niet meer te doen. Om dit alles in goede banen te leiden zijn ze op zoek naar een ondersteunend programma waarmee ze deze activiteiten kunnen plannen. Hierbij moet rekening gehouden worden met de beschikbaarheid, deskundigheid en woonplaats van de docenten van Scala, de roosterwensen van de scholen en diverse randvoorwaarden, als de volgorde van schoolbezoeken. We hadden contact met Scala via Maarten Gal die voorgaande jaren altijd met de hand het rooster heeft gemaakt.

2. LITERATUURONDERZOEK

Optimalisatie is een tak van de wiskunde die zich bezighoudt met het vinden van de beste oplossing voor een bepaald probleem. Dit wordt gedaan door het probleem uit te drukken in een wiskundige notatie om zo een wiskundig model te creëren dat het probleem beschrijft. Er zijn veel gebieden waar optimalisatie vaak wordt gebruikt, bijvoorbeeld in productie, waarbij het doel kan zijn om de kosten van het maken van een bepaald product te minimaliseren of om de meest efficiënte route te vinden voor leveringen naar bepaalde bestemmingen.

Een optimalisatieprobleem dat op een groot aantal gebieden voorkomt, is het maken van een rooster. Het doel is om een zo goed mogelijk rooster te vinden dat niet alleen mogelijk is praktisch gezien, maar ook voldoet aan de voorkeuren van alle betrokken partijen. Hoewel optimalisatie een effectieve techniek is om de beste oplossing voor een verscheidenheid aan problemen te vinden, zijn sommige problemen erg groot en kan het vinden van een oplossing zeer tijdrovend zijn.

Er zijn heel veel verschillende roosterproblemen. Bijvoorbeeld een rooster maken voor een universiteit waarin alle cursussen zijn gepland of voor een luchtvaartmaatschappij met alle vluchten. Andrea Schaerf beschrijft in zijn artikel *A Survey of Automated Timetabling* (1991) drie soorten roosterproblemen: een rooster voor een school, een lesrooster voor de universiteit en een examenrooster. Alle drie de soorten roosters lijken op elkaar; in alle drie de situaties moet een klas aan een docent en een lokaal gekoppeld worden. Echter, toch zijn er genoeg verschillen in de situaties die zorgen dat het echt verschillende roosterproblemen zijn. Zo is een verschil dat bij de school een klas vast staat, terwijl bij een universiteit studenten van verschillende studies bij elkaar mogen komen. Ook mag een klas op school absoluut niet twee lessen op hetzelfde moment hebben, terwijl dit op een universiteit niet wenselijk is maar soms niet anders kan. Daarentegen is het bij examens wel weer zo dat studenten absoluut niet twee examens tegelijkertijd mogen hebben. Elke cursus op de universiteit heeft meestal ook maar één examenmoment, terwijl er in de periode die er vooraf gaat wel meerdere colleges zijn. Een klas op een school heeft vaak één of meerdere keren per week les in een bepaald vak gedurende het hele jaar. Deze verschillen zorgen ervoor dat de drie situaties verschillende constraints hebben en dus ook voor verschillende modellen en uitkomsten zorgen.

Het roosterprobleem van Scala heeft op een aantal punten overeenkomsten met de drie hierboven

besproken roosterproblemen. Zo moet een docent ook aan een groep gekoppeld worden en mogen zowel groepen als docent niet meerdere workshops op hetzelfde moment hebben/geven. Er hoeft daarentegen geen rekening gehouden te worden met de lokatie. De docent gaat naar de school toe en er wordt verwacht dat er daar ruimte is om de workshop te geven. Een punt dat ook anders is dan een school- of universiteitrooster is dat bij dit roosterprobleem rekening gehouden moet worden met reistijden. Zo heeft een docent een kwartier reistijd tussen verschillende scholen in dezelfde plaats en een half uur tussen verschillende scholen in verschillende plaatsen. Een docent moet op een dag wel in één gemeente blijven. Zo lijkt dit roosterprobleem op bepaalde punten op andere roosterproblemen, maar door zijn eigen specifieke eisen is het toch een heel ander probleem geworden.

Dit verslag beschrijft hoe het roostervraagstuk van Scala voor het plannen van de gevraagde workshops op verschillende scholen kan worden opgelost door gebruik te maken van integer programmeren. Integer programmeren is een gangbare aanpak om dit soort problemen op te lossen. Zo lossen Havas, Olsson, Persson en Schierscher (2013) het roostervraagstuk voor een aantal vakken van de afdeling Wiskundige Wetenschappen van de Universiteit van Göteborg en Chalmers University of Technology hiermee op. Ryan en Foster (1981) passen dit toe op een metropolitische bus operatie en Dimopoulou en Miliotis (2001) maken hier gebruik van bij het maken van een gecombineerd universitair cursus-examen rooster. Dit zijn slechts enkele voorbeelden. Het rooster van Scala wordt momenteel nog met de hand gemaakt. Dit kost veel tijd. Door een geautomatiseerd programma zal het proces veel sneller gaan. Dit programma is gebaseerd op een integer linear programming (ILP) model en zal daardoor proberen een oplossing te vinden. Deze manier om het roosterprobleem op te lossen is de exacte methode. Het roosterprobleem zou ook door een heuristische methode opgelost kunnen worden.

3. PROBLEEMORIËNTATIE

Elk jaar voor de zomervakantie stuurt Scala naar alle scholen waarmee ze verbonden zijn een lijst met de workshops die het volgende schooljaar gegeven gaan worden. De scholen mogen vervolgens voor elke groep kiezen welke workshop(s) ze dat jaar willen krijgen. Als al deze gegevens binnen zijn kan er begonnen worden met het maken van een rooster waarin een docent en een dag met tijdstip wordt verbonden aan de gevraagde workshops voor elke groep. Belangrijk is dat elke workshop die gevraagd wordt daadwerkelijk gegeven gaat worden in het schooljaar. Daarnaast moeten de workshops ook alleen gepland worden op dagen en op tijden dat de groep kan én de bijbehorende docent kan. Ook kan een workshop een afwezigheidsperiode hebben, wat inhoudt dat die workshop in die periode niet gegeven mag worden. Een docent kan ook niet meer lessen geven dan hij/zij tijd heeft op die dag om te werken. De afwezigheidsperiode van de groepen, docenten en workshops en de beschikbare tijden van de groepen en docenten kunnen allemaal aangegeven worden in het Excelbestand. Naast het plannen op de beschikbare tijden komen er nog meer zaken bij kijken waar rekening mee gehouden moet worden tijdens het plannen. Zo moeten er minimaal 30 dagen zitten tussen verschillende workshops van dezelfde groep. Echter, er bestaan ook workshops die bestaan uit vier lessen, de zogenoemde lessenseries. Deze lessen uit de lessenseries moeten vier maal achter elkaar elke week op dezelfde dag worden gegeven.

Scala biedt workshops aan van verschillende kunstdisciplines: muziek, theater, dans, media, beeldend en erfgoed. Alle docenten zijn verbonden aan één discipline. Docenten mogen dus alleen workshops geven van die specifieke discipline. Achter elke workshop in het Excelbestand

is ruimte beschikbaar om specifieke docenten op te geven die die workshop mogen geven. Als er niks wordt ingevuld betekent dit dat alle docenten van de discipline die bij die workshop horen deze workshop mogen geven. Als er wel docenten worden ingevuld zijn er nog twee gevallen te onderscheiden: met of zonder prioriteit. Als het niet uitmaakt welke docent van de toegewezen docenten deze workshop geeft komen ze allemaal bij de eerste prioriteit te staan. Zijn er wel verschillen in prioriteit tussen de toegewezen docenten, dan wordt aan elke docent een prioriteiten nummer gegeven. Uiteindelijk is het doel om zoveel mogelijk docenten met de hoogste prioriteit in te plannen. De doelfunctie zal hiervoor zorgen. Voor de disciplines media en beeldend bestaan nog speciale voorwaarden waar het model aan moet voldoen. Doordat de docenten van de discipline beeldend veel materialen mee moeten nemen, kunnen zij maximaal drie workshops op een dag geven. Bij de workshops van de discipline media zijn materialen nodig waar niet veel van aanwezig zijn. Zodoende kunnen maximaal twee docenten van de discipline media op dezelfde dag aan het werk zijn.

Naast workshops op het reguliere onderwijs biedt Scala ook de mogelijkheid om workshops te krijgen op het speciaal onderwijs. Scala heeft een aantal docenten die bevoegd zijn om les te geven op het speciaal onderwijs. Alleen deze docenten mogen dus lessen geven aan deze scholen.

Als een docent op dezelfde dag op verschillende scholen workshops moet geven, dan zit daar reistijd tussen. Er is afgesproken dat als de docent naar een andere school gaat, maar wel in dezelfde plaats blijft er een kwartier reistijd geldt. Gaat de docent naar een andere school die ook in een andere plaats staat, dan moet er een half uur reistijd gerekend worden. Een docent mag niet op een dag naar scholen in verschillende gemeentes.

Het doel van deze bacheloropdracht is om een programma te maken die, als de data op de juiste manier wordt ingevoerd, een rooster voor de culturele instelling Scala genereert. Dit programma is gemaakt in het programmeerprogramma C#. Met deze bacheloropdracht wordt dus onderzocht hoe het roosterprobleem van Scala op een exacte manier is op te lossen.

3.1. Constraints

Om duidelijke randvoorwaarden te hebben om een wiskundig model te maken, zijn alle punten waar rekening mee gehouden moet worden opgedeeld in hard constraints en soft constraints. De constraints zijn geformuleerd op basis van informatie verkregen van Maarten Gal. Hij was de afgelopen jaren verantwoordelijk voor het maken van het rooster bij Scala. De hard constraints zijn nodig om een geldig tijdschema te krijgen en de soft constraints weerspiegelen de voorkeuren. Als er ** achter de constraint staat betekent dit dat deze constraint wordt meegenomen in de implementatie, bij *nummer zal het gedeeltelijk meegenomen worden (in paragraaf 3.1.3 zal uitgelegd worden op welke manier). Als er niks achter de constraint staat zal deze constraint nog niet meegenomen worden in de huidige implementatie. Dit zijn punten die in een volgend model meegenomen kunnen worden om het model te verbeteren.

3.1.1 Hard constraints

- Al de aangevraagde workshops moeten geroosterd worden in het aangegeven tijdsinterval (één schooljaar). **
- Workshops die bestaan uit meerdere lessen, de lessenseries, moeten in achtereenvolgende weken op dezelfde dag worden gegeven door dezelfde docent. **

- Een groep mag niet meer dan één workshop per 31 dagen hebben met uitzondering van de lessenseries. ^{*1}
- Een workshop mag alleen gegeven worden in de periode die ervoor staat. **
- Elke workshop heeft een voorgeschreven tijdsduur die zonder onderbrekingen gegeven moet worden. **
- Groepen kunnen alleen workshops krijgen op de tijden dat de groep kan. **
- Docenten geven alleen de workshops die Scala aangeeft dat ze mogen doen. **
- Groepen in het speciaal onderwijs kunnen alleen workshops krijgen van docenten die gespecialiseerd zijn in het speciaal onderwijs. ^{*2}
- Docenten kunnen alleen workshops geven op de tijden dat ze beschikbaar zijn. **
- Een docent kan niet meer dan één workshop tegelijkertijd geven. ^{*3}
- Docenten mogen op een dag niet in verschillende gemeentes werken. **
- Een docent van de discipline beeldend kan maximaal drie workshops beeldend op een dag geven (door vervoer van het materiaal). **
- Er kunnen maximaal twee docenten media op dezelfde dag aan het werk zijn (door te weinig materiaal). **
- Tussen twee opeenvolgende workshops op een dag van één docent in verschillende plaatsen moet minimaal een half uur tijd zitten om te reizen. Tussen verschillende scholen in dezelfde plaats is dit een kwartier tijd.

3.1.2 Soft constraints

- Workshops die bestaan uit meerdere lessen worden in achtereenvolgende weken precies op hetzelfde moment gegeven.
- Docenten moeten geen of zoveel mogelijk workshops geven op een dag.
- Het indelen van docenten op workshops gebeurt op basis van de voorkeuren van Scala. **

3.1.3 Toelichting gedeeltelijk meegenomen constraints

Zoals hierboven genoemd zijn er een aantal constraints gedeeltelijk meegenomen in het model. Hieronder is toegelicht waarom en op welke manier deze constraints gedeeltelijk zijn meegenomen.

- ^{*1} In de constraints die zijn geïmplementeerd in het model is een constraint die zegt dat tussen verschillende workshops van een groep ten minste 30 dagen moeten zitten. Echter, dit gaat alleen over alle eerste lessen van de workshops. Dus bij een lessenserie en een workshop die eenmalig is, kan het voorkomen dat tussen de laatste les van de lessenserie en de eenmalige workshop maar negen dagen zitten.
- ^{*2} Er staat een kolom voor speciaal onderwijs in de data. Met het implementeren is hier geen rekening mee gehouden. Door dit punt echter wel mee te nemen is het de bedoeling dat in de data de workshops voor speciaal onderwijs apart genoemd worden met daarachter alleen de docenten die deze workshops voor speciaal onderwijs mogen geven.

- ^{*3} Er worden in het model intersecties gemaakt van de aanwezigheidstijden van de groep en de werktijden van de docent die de workshop aan die groep gaat geven. Er wordt dan naar elke intersectie op een dag gekeken of er genoeg tijd is om de betreffende workshop te geven. Dit zorgt ervoor dat de workshop zonder onderbrekingen gegeven kan worden en groepen de workshop krijgen en docenten de workshop geven op tijden dat ze beschikbaar zijn. In dit model wordt alleen nog niet gewerkt met precieze tijden. Daarom kan niet een specifieke tijd gereserveerd worden voor een groep. Daardoor is het dus mogelijk dat er overlap is voor een docent tussen verschillende workshops op een dag en kan het voorkomen dat een docent uiteindelijk toch op twee plekken tegelijkertijd moet zijn. Dit geldt bijvoorbeeld als twee groepen van 9:00 tot 10:00 uur een workshop kunnen krijgen, de docent werkt van 9:00 tot 12:00 uur en beide groepen een workshop hebben aangevraagd van 60 minuten. Volgens de intersecties van de tijden kunnen allebei de groepen een workshop van een uur krijgen en twee uur workshops geven is mogelijk binnen de drie uur dat de docent werkt. Het programma ziet op dit moment nog niet dat de twee workshops dan tegelijkertijd zijn.

3.2. Aannames

Aangezien dit roosterprobleem veel randvoorwaarden kent, zijn er voor de implementatie een aantal aannames gedaan:

- In de toegestuurde data van Scala stond dat docenten voor elke dag op een ochtend en middag inzetbaar, incidenteel inzetbaar of niet inzetbaar konden zijn. Aangezien incidenteel inzetbaar een erg ruim en vaag begrip is, is er voor gekozen dat docenten die incidenteel inzetbaar zijn niet inzetbaar zijn.
- De ochtend loopt van 9 uur tot 12 uur. De middag van 12 uur tot 15 uur. Er is per dagdeel dat een docent werkt een half uur afgehaald om te gebruiken als reistijd. Alle scholen beginnen tussen 8:15 uur en 8:45 uur. Zodoende is er voor gekozen dat docenten om 9:00 uur beginnen zodat alle groepen de dag kunnen openen zonder direct aan een workshop te beginnen. Als de docent om 8:00 uur zou beginnen zou de docent een uur meer tijd hebben om te werken terwijl er nog haast geen klassen aanwezig zijn. Daardoor zal er vaker voorkomen dat een docent meerdere workshops tegelijkertijd moet geven, zoals besproken is bij punt ^{*3} in bovenstaande paragraaf over gedeeltelijk meegenomen constraints.
- Als een docent maar een deel van de ochtend/middag beschikbaar is, dan is die docent de hele ochtend/middag niet beschikbaar.
- In het bestand met alle gegevens van Scala stond soms dat sommige groepen gesplitst moeten worden. Op advies van Maarten Gal is dit verwaarloosd.
- Aangezien er nog geen gegevens zijn van in welke gemeente elke docent woont, is dit gegeven verwaarloosd. Hierdoor wordt er niet meegenomen dat een docent in de dichtstbijzijnde gemeente werkt om reiskosten te verminderen.
- Daarnaast zijn de voorkeuren van in welke gemeentes docenten niet willen werken verwaarloosd. Het is vanuit Scala nog niet helemaal duidelijk of dit punt wel of niet meegenomen moet worden.
- Een groep vraagt niet in een jaar hetzelfde project aan.
- Docenten hebben één discipline.

- Er bestonden in de gegevens van Scala drie workshops met als discipline multi. Aangezien aan deze workshops geen docenten waren gekoppeld en er geen docenten waren met discipline multi, zijn die workshops verwijderd uit het aanbod. Die workshops zouden immers zonder docenten niet gegeven kunnen worden.
- Reistijden worden in dit model niet meegenomen, maar om toch gedeeltelijk hiermee rekening te houden is er een half uur werktijd afgehaald per dagdeel. Als de docent een hele dag werkt (9:00 tot 15:00 uur) zijn er dus vijf uren beschikbaar om te werken en een uur om te reizen.
- Op aanraden van Maarten Gal is er geen rekening gehouden met pauzes voor de docent.
- Zoals eerder genoemd heeft dit model nog geen precieze tijdsindeling. Er is nu alleen gepland op dagniveau. Dit was in eerste instantie een versimpeling van het probleem voor een eerste model. Door tijdsgebrek heeft dit punt zich niet meer verder kunnen ontwikkelen.
- Bij de workshops mogen er alleen meerdere docenten de hoogste prioriteit hebben. Bij de lagere prioriteiten mogen slechts één docent staan.

4. MODEL

Deze paragraaf bevat de wiskundige beschrijving van het probleem en een lijst van alle opgenomen verzamelingen, de variabelen, de doelfunctie en de constraints. Het model is zo gemaakt dat het rooster afhangt van welke data er ingevoerd wordt. Zolang de data maar op de juiste manier in het Excelbestand staat (zie hoofdstuk 10.2), wordt er een specifiek rooster gemaakt voor die data.

4.1. Verzamelingen

- $A_{l,t,d} = \{\text{intersectiesaanwezigheidstijden}\}$
De intersecties van de aanwezigheidstijden van een lesactiviteit (dus wanneer een workshop aan een groep gegeven kan worden) en een docent op een dag.
- $D = \{0, 1, \dots, 312, 313\}$
De dagen van het schooljaar waarin gepland gaat worden.
- $G = \{\text{groepen}\}$
Alle groepen die workshops krijgen.
- $L = \{\text{lesactiviteiten}\}$
Alle lesactiviteiten waarbij groepen aan hun gevraagde workshops gekoppeld zijn.
- $L_{1,g} = \{\text{eerstelessen}\}$
Alle eerste lessen van de workshops van één groep.
- $L_{\text{beeldend}} = \{\text{lesactiviteitenbeeldend}\}$
Alle lesactiviteiten van de discipline beeldend.
- $L_4 = \{\text{lesactiviteitenviermaal}\}$
Lesactiviteiten van een workshop dat vier maal gegeven moet worden.
- $L_g = \{\text{lesactiviteitengroep}\}$
De lesactiviteiten van groep g .

- $L_{gem} = \{lesactiviteitengemeente\}$
Alle lesactiviteiten in een bepaalde gemeente.
- $P_l = \{45, 60, 90, 120\}$
Duur van les l .
- $P_{d,t} = \{0, 150, 300\}$
Beschikbare tijd om te werken van docent t op dag d .
- $P_l^n = \{prioriteitenlijst\}$
Van een lesactiviteit zitten in deze lijst alle docenten die deze lesactiviteit mogen geven met de n^{de} prioriteit. Prioriteit 1 heeft de hoogste prioriteit, prioriteit 8 de laagste.
- $T = \{docenten\}$
Alle docenten die workshops geven.
- $T_{media} = \{docentenmedia\}$
De docenten van de discipline media.

4.2. Variabelen

- $x_{l,t,d}$
Deze binaire variabele geeft aan of een bepaalde lesactiviteit l gekoppeld is aan een bepaalde docent t op een bepaalde dag d . Als deze variabele 1 is dan is het wel gekoppeld, bij 0 is dit niet zo. $x_{l_n,t,d}$ gaat over de lesactiviteit l met serienummer n . Als er geen serienummer staat gaat het over een lesactiviteit van een workshop die slechts één keer gegeven wordt. Dit kan worden gezien als $x_{l_0,t,d}$.
- $w_{t,d}$
Deze binaire variabele geeft aan of een bepaalde docent t wel (1) of niet (0) werkt op dag d .

4.3. Doelfunctie

De constraints zorgen er altijd voor dat het resulterende rooster klopt aan alle voorwaarden die er gesteld zijn. De doelfunctie is er om voorkeuren te verwerken in het programma. Zo moet in dit roosterprobleem de doelfunctie rekening houden met de voorkeuren van docenten voor de verschillende workshops. Dit wordt gedaan door gewichten toe te kennen aan de variabele $x_{l,t,d}$. Als een lesactiviteit gekoppeld wordt aan een docent wordt de bijbehorende variabele automatisch toegevoegd aan de doelfunctie met de coëfficiënt $g_{l,t}$ die afhangt van in welke prioriteitenlijst de docent staat. De doelfunctie wordt geminimaliseerd, dus bij een docent uit de hoogste prioriteitenlijst (prioriteit 1) wordt de coëfficiënt negatiever dan bij docenten met lagere prioriteiten. Bij een docent uit prioriteitenlijst 1 is een coëfficiënt van -50 verbonden, bij prioriteitenlijst 2 -20, bij prioriteitenlijst 3 -10, bij prioriteitenlijst 4 -5, bij prioriteitenlijst 5 -4, bij prioriteitenlijst 6 -3, bij prioriteitenlijst 7 -2 en bij prioriteitenlijst 8 -1. Door te sommeren over alle dagen, docenten en lesactiviteiten en dit te minimaliseren haalt het programma een zo optimaal mogelijke oplossing voor het probleem, wat inhoudt dat docenten met hoge prioriteit eerder gekozen worden dan lagere prioriteit. Het liefst zelfs allemaal uit prioriteitenlijst 1. De doelfunctie

ziet er als volgt uit:

$$\min \sum_{t \in T} \sum_{d \in D} \sum_{l \in L} x_{l,t,d} \cdot g_{l,t}$$

4.4. Constraints

Hieronder zijn de constraints wiskundig weergegeven. Onder elke constraint staat uitgelegd wat die constraint precies betekent. Aangezien de solver die in het programma gebruikt wordt gebruik maakt van een boven- en ondergrens, zijn de constraints hier ook op deze wijze weergegeven.

$$\sum_{t \in T} \sum_{d \in D} x_{l,t,d} = 1, \quad \forall l \in L \quad (1)$$

(1) Alle lesactiviteiten moeten worden gekoppeld aan een docent en een dag.

$$x_{l,t,d} = 0, \quad \forall l \in L, \forall d \in D, \forall t \notin P_l^n \text{ met } n = 1, \dots, 8 : \quad (2)$$

(2) Als een bepaalde docent deze les niet mag geven, moet hij niet gekoppeld worden. Ofwel, als de docent niet in één van de prioriteitenlijsten zit van een lesactiviteit, is de variabele $x_{l,t,d}$ direct nul.

$$0 \leq \sum_{l \in L} x_{l,t,d} \cdot P_l \leq P_{d,t}, \quad \forall t \in T, \forall d \in D \quad (3)$$

(3) De som van de duur van de lesactiviteiten die een docent op een dag geeft mag niet meer zijn dan zijn/haar werktijden.

$$0 \leq \sum_{l \in L_{1,g}} \sum_{t \in T} \sum_d^{d+30} x_{l,t,d} \leq 1, \quad \forall d \in D, \forall g \in G \quad (4)$$

(4) Tussen verschillende workshops van een groep moeten ten minste 30 dagen zitten.

$$x_{l_0,t,d} - x_{l_n,t,d+7 \cdot n} = 0, \quad \text{voor } n = 1, 2, 3, \forall l \in L_A, \forall t \in T, \forall d \in D \quad (5)$$

(5) Workshops uit een lessenserie moeten vier maal achter elkaar elke week op dezelfde dag worden ingepland met dezelfde docent.

$$0 \leq x_{l,t,d} \cdot P_l \leq a, \quad a \in A_{l,t,d}, \forall l \in L_g, \forall g \in G, \forall t \in T, \forall d \in D \quad (6)$$

(6) Een workshop kan niet gegeven worden aan een groep als de duur van de workshop groter is dan de intersectie van beschikbaarheid op een dag van een groep, docent en workshop.

$$0 \leq \sum_{l \in L} x_{l,t,d} - w_{t,d} \leq \infty, \quad \forall t \in T, \forall d \in D \quad (7)$$

(7) Als een docent geen lesactiviteit gepland heeft op een dag dan is hij/zij niet aan het werk op die dag.

$$-1 \leq x_{l,t,d} - w_{t,d} \leq 0, \quad \forall l \in L, \forall t \in T, \forall d \in D \quad (8)$$

(8) Als een docent een lesactiviteit gepland heeft op een dag dan is hij/zij aan het werk op die dag.

$$0 \leq \sum_{t \in T_{media}} w_{t,d} \leq 2, \quad \forall d \in D \quad (9)$$

(9) Er kunnen maximaal twee docenten media tegelijkertijd op een dag aan het werk zijn.

$$0 \leq \sum_{l \in L_{beeldend}} x_{l,t,d} \leq 3, \quad \forall t \in T, \forall d \in D \quad (10)$$

(10) Een docent van de discipline beeldend kan maar drie lesactiviteiten beeldend geven op één dag.

$$0 \leq x_{la,t,d} + x_{lb,t,d} \leq 1, \quad la \in L_{gem}, lb \notin L_{gem} \forall gemeentes, \forall t \in T, \forall d \in D \quad (11)$$

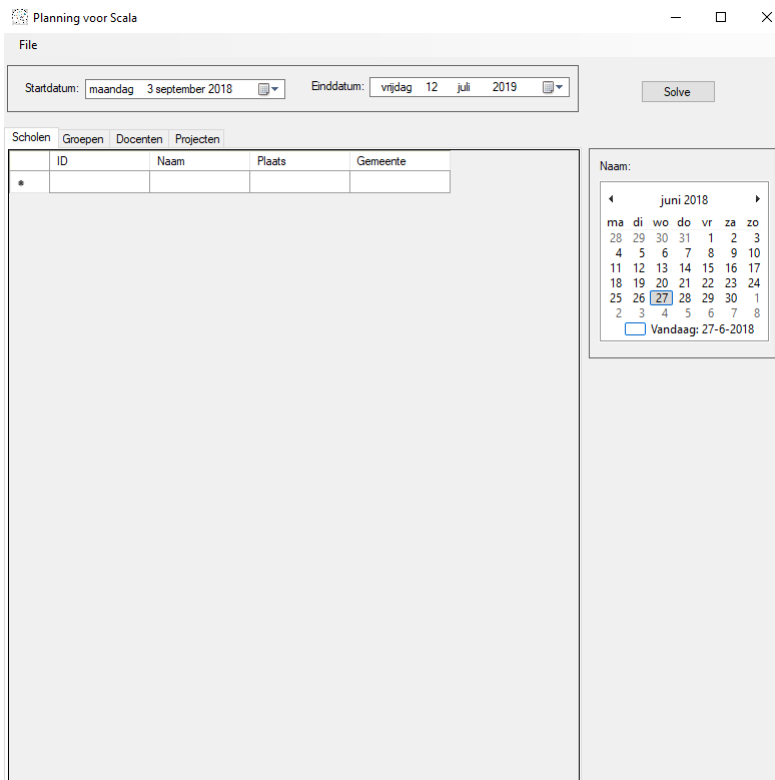
(11) Een docent mag op één dag maar in één gemeente workshops geven.

5. RESULTATEN

In dit hoofdstuk is eerst een toelichting gegeven bij het programma. Hier is benoemd wat de stappen zijn om het programma te gebruiken om uiteindelijk een rooster te krijgen en is een beeld gegeven van hoe het programma eruit ziet. In de paragraaf daarna is nagegaan of alle constraints in het programma werken en zijn er voorbeelden gegeven waaraan dit te zien is.

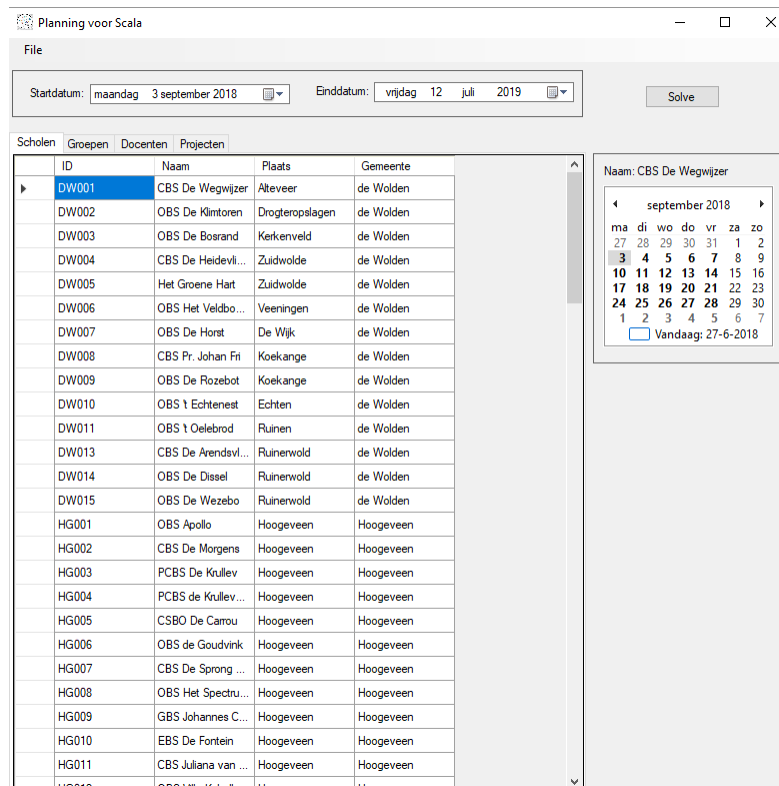
5.1. Toelichting programma

Het programma is geschreven in C# en bestaat uit ruim 2000 regels (de code van de MILP solver niet meegerekend). Tijdens de eerste fase van het programma worden alle gegevens vanuit Excel geïmporteerd. Hoe het programma eruit ziet voor de gebruiker is weergegeven in Figuur 1.



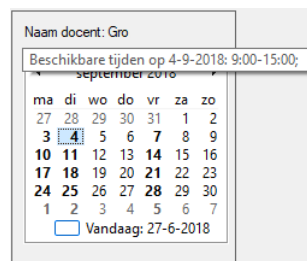
Figuur 1: Programma als het geopend wordt.

De eerste stap is de startdatum en einddatum aangeven, waartussen een rooster gemaakt moet worden. Door op *File* en vervolgens op *Open excel bestand ...* te klikken, kan in de computer het juiste Excelbestand gezocht worden om in het programma te laden. Als het juiste bestand gekozen is gaat het programma alle gegevens importeren. Deze gegevens verschijnen na het importeren in het programma, zoals te zien is in Figuur 2.



Figuur 2: Programma met geïmporteerde gegevens.

Bij de kalender aan de rechterkant is te zien welke dagen een school, groep, docent of workshop aanwezig is, dan is de datum dikgedrukt. Als de school, groep, docent of workshop niet aanwezig is op een dag, dan is de datum niet dikgedrukt. Bij een groep en docent is er ook de mogelijkheid om precies te kijken op welke tijden ze aanwezig zijn door op de desbetreffende datum te klikken. Voor docent Gro is hieronder in Figuur 3 weergegeven dat zij op dinsdag 4 september beschikbaar is van 9:00 tot 15:00 uur.



Figuur 3: De aanwezigheid van docent Gro in september 2018.

Nu alle gegevens zijn geïmporteerd is het tijd voor de volgende fase: een rooster genereren. De gebruiker hoeft hiervoor alleen op de knop "solve" te drukken. Het programma gaat dan, zonder dat daar iets van te zien is, aan het werk om het optimaalste rooster te vinden. Het programma gebruikt hierbij een externe solver: CPLEX Optimizer (in het programma terug te

vinden als de mixed-integer linear programming (MILP) solver). Deze solver is in staat integer linear programming problemen, zoals dit roosterprobleem, op te lossen. Als het programma uitgerekend is, wordt er automatisch een Excelbestand geopend met daarin het rooster. Een voorbeeld van hoe dit eruit ziet is gegeven in Figuur 4.

	A	B	C	D	E	F	G	H	I
1	objective		-300						
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3		1 DW001	1,2	de Wolden	BE01	beeldend	Bar	92	4-12-2018
4		2 DW005	1,2a	de Wolden	DA01	dans	Slo	0	3-9-2018
5		3 HG002	1a	Hoogeveen	ME01	media	Ach	2	5-9-2018
6		4 HG008	4	Hoogeveen	MU01	muziek	Bre	0	3-9-2018
7		5 MP006	SO2	Meppel	TH01	theater	Eba	1	4-9-2018
8		6 SL009	3	Steenwijkerland	ER01	erfgoed	Wie	0	3-9-2018

Figuur 4: Voorbeeld rooster geëxporteerd naar een Excelbestand.

In dit voorbeeld zijn er zes workshops aangevraagd, waar geen lessenseries tussen zitten. Zodoende zijn er zes lesactiviteiten. Naast elke lesactiviteit is weergegeven over welke school en groep het gaat, in welke gemeente de school staat, welke workshop is ingepland, van welke discipline deze workshop is, welke docent aan de lesactiviteit gekoppeld is en op welke dag de lesactiviteit is gepland. Dag 0 is de eerste dag dat gepland mag worden. In dit geval is dat 3 september 2018, zie Figuur 2. Vervolgens wordt elke dag als een dag geteld, dus ook de zaterdagen en zondagen tellen mee in de hoeveelheid dagen. De doelfunctie (hier aangeduid als "objective") heeft in deze situatie een waarde van -300. Er waren zes lesactiviteiten, wat betekent dat bij elke lesactiviteit een docent uit de hoogste prioriteitenlijst is gekozen, aangezien de waarde die daarbij hoort -50 is.

5.2. Constraints controleren

Van totaal 106 scholen, die samen 717 groepen hebben, zijn er gegevens verkregen. Er zijn een aantal scholen en groepen uit gehaald waarvan geen gegevens waren, aangezien voor die groepen geen workshops gepland kunnen worden. Elke groep kan maximaal vier workshops kiezen per jaar uit een lijst met precies 100 workshops. Daarnaast zijn er in totaal 56 docenten die workshops geven. De periode waarbinnen geroosterd moet worden loopt van 3 september 2018 tot en met 12 juli 2019. Dit zijn 313 dagen die het programma steeds afloopt. Het programma koppelt aan elke workshop van een groep een docent en plant dit op een bepaalde dag. Zo ontstaan er maximaal $4 \cdot 717 \cdot 56 \cdot 313 = 50.270.304$ variabelen die het programma moet berekenen. Dit waren veel te veel variabelen voor de computer waarop het programma werd gerund, waardoor al snel duidelijk werd dat er hiervoor onvoldoende geheugen beschikbaar was. Daardoor was het onvermijdelijk dat het programma alleen getest kon worden met kleinere Excelbestanden met minder workshops die gepland moesten worden.

Daarnaast heeft Scala helaas niet een bestand met alle gevraagde workshops aangeleverd voor het einde van de bacheloropdracht. Daarom was het ook niet mogelijk om het programma te testen op de echte gegevens waarop het programma een rooster zou moeten maken. Zodoende zijn er geen resultaten die overeen komen met de werkelijkheid. Wel zijn er bestanden gemaakt waarmee is getest of alle constraints op de juiste manier werken. Deze constraints worden hieronder één voor één besproken met een ondersteunend voorbeeld erbij van het geëxporteerde rooster in Excel.

Alle constraints zijn met veel verschillende datasets getest. Bij elke constraint is slechts één van de resultaten weergegeven waarmee geverifieerd kan worden dat de constraint het gewenste resultaat geven. Wegens privacyredenen worden de docenten niet bij hun volledige naam genoemd.

Constraint 1: Alle lesactiviteiten moeten worden gekoppeld aan een docent en een dag.

Dat deze constraint werkt is in het Excelbestand met het rooster goed te zien aan zowel dat de objective niet nul is als dat er achter elke lesactiviteit een docent en een dag staat (zie Figuur 4). Als er geen rooster te maken is, dan wordt de objective nul en staat er achter geen enkele lesactiviteit een docent en een dag. Dit komt bijvoorbeeld voor in een situatie wanneer er meer workshops gepland zijn dan de docenten die die workshops mogen geven aankunnen in de tijdsperiode die ervoor staat (zie constraint 2 en 3).

Constraint 2: Als een bepaalde docent deze les niet mag geven, moet hij niet gekoppeld worden. Ofwel, als de docent niet in één van de prioriteitenlijsten zit van een lesactiviteit, is de variabele $x_{l,t,d}$ direct nul.

Bij alle uitkomsten in de Excelbestanden, zoals Figuur 4, is te controleren dat de docent die gekoppeld is aan de lesactiviteit komt uit een prioriteitenlijst van de betreffende workshop. Aan de waarde van de objective valt af te leiden of alle docenten uit de hoogste prioriteitenlijst komen. Als het aantal lesactiviteiten vermenigvuldigd met -50 uitkomt op de waarde van de objective, dan komen alle docenten uit de hoogste prioriteitenlijst. Als dit niet het geval is, dan is er minstens één docent die uit een lagere prioriteitenlijst komt. In de situatie van de roosters die hieronder zijn weergegeven in Figuur 5 en Figuur 6 moest workshop BE02 zes en zeven keer, respectievelijk, gegeven worden op 5 september. In de eerste prioriteitenlijst van deze workshop staat docent Kam en in de tweede prioriteitenlijst docent Lee. Aangezien beide docenten de hele dag werkten (vijf uur) en de workshop 90 minuten duurt, kunnen beide docenten op één dag maximaal drie workshops BE02 geven. In Figuur 5 is te zien dat allebei de docenten precies drie workshops BE02 geven op 5 september. In de tweede situatie moesten zeven workshops gepland worden. Echter dit kunnen deze twee docenten niet aan op één dag. Het programma pakt dan niet zomaar een andere docent die wel kan, aangezien deze constraint ervoor zorgt dat alleen docent Kam en docent Lee deze workshop mogen geven. Doordat het programma daardoor niet meer kan voldoen aan constraint 1 wordt de objective nul en worden de lesactiviteiten niet meer gekoppeld aan een docent en een dag, zoals te zien is in Figuur 6.

	A	B	C	D	E	F	G	H	I
1	objective	-210							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	HG001	3	Hoogeveen	BE02	beeldend	Lee	0	5-9-2018
4	2	HG002	4,5a	Hoogeveen	BE02	beeldend	Lee	0	5-9-2018
5	3	HG003	6,7,8a	Hoogeveen	BE02	beeldend	Lee	0	5-9-2018
6	4	HG006	1	Hoogeveen	BE02	beeldend	Kam	0	5-9-2018
7	5	HG006	7,8	Hoogeveen	BE02	beeldend	Kam	0	5-9-2018
8	6	HG007	6a	Hoogeveen	BE02	beeldend	Kam	0	5-9-2018

Figuur 5: Zes workshops op één dag voor twee docenten.

	A	B	C	D	E	F	G	H
1	objective	0						
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag
3		1 HG001	3	Hoogeveen	BE02	beeldend		
4		2 HG002	4,5a	Hoogeveen	BE02	beeldend		
5		3 HG003	6,7,8a	Hoogeveen	BE02	beeldend		
6		4 HG006	1	Hoogeveen	BE02	beeldend		
7		5 HG006	7,8	Hoogeveen	BE02	beeldend		
8		6 HG007	6a	Hoogeveen	BE02	beeldend		
9		7 HG008	6,7	Hoogeveen	BE02	beeldend		

Figuur 6: Zeven workshops op één dag voor twee docenten.

Constraint 3: De som van de duur van de lesactiviteiten die een docent op een dag geeft mag niet meer zijn dan zijn/haar werktijden.

Om te laten zien dat deze constraint werkt, zijn er twee situaties geroosterd: één waarbij de situatie nog wel oplosbaar is en de ander waarbij de situatie door constraint drie niet meer oplosbaar is. De workshop met ID MU06 mag alleen gegeven worden door docent Dij. In deze situatie hebben zeven groepen deze workshop aangevraagd welke allemaal gegeven moeten worden op 3 september 2018 of 4 september 2018. Docent Dij is op allebei de dagen de hele dag aanwezig, wat betekent dat hij beide dagen vijf uur kan werken. Workshop MU06 is een eenmalige workshop van 60 minuten. Dit betekent dat docent Dij beide dagen maximaal vijf workshops MU06 kan geven en in deze twee dagen dus maximaal tien workshops, aangezien de scholen ook allemaal in dezelfde gemeente liggen. In Figuur 7 is de situatie te zien met tien workshops. Deze situatie is net oplosbaar doordat de docent elke dag vijf workshops kan geven. In de situatie van Figuur 8 is er nog één workshop bijgekomen die ervoor zorgt dat er geen rooster meer te maken is die aan deze constraint voldoet. Zodoende werd de objective nul en werden er geen docent en dagen meer gekoppeld aan de lesactiviteiten.

	A	B	C	D	E	F	G	H	I
1	objective	-500							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3		1 DW001	1,2	de Wolden	MU06	muziek	Dij	1	4-9-2018
4		2 DW002	4,5,6	de Wolden	MU06	muziek	Dij	1	4-9-2018
5		3 DW004	5	de Wolden	MU06	muziek	Dij	1	4-9-2018
6		4 DW005	1,2c	de Wolden	MU06	muziek	Dij	1	4-9-2018
7		5 DW005	6b	de Wolden	MU06	muziek	Dij	1	4-9-2018
8		6 DW006	1,2	de Wolden	MU06	muziek	Dij	0	3-9-2018
9		7 DW007	4	de Wolden	MU06	muziek	Dij	0	3-9-2018
10		8 DW008	5,6	de Wolden	MU06	muziek	Dij	0	3-9-2018
11		9 DW011	3	de Wolden	MU06	muziek	Dij	0	3-9-2018
12		10 DW013	4	de Wolden	MU06	muziek	Dij	0	3-9-2018

Figuur 7: Tien workshops in twee dagen voor één docent.

	A	B	C	D	E	F	G	H
1	objective	0						
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag
3		1 DW001	1,2	de Wolden	MU06	muziek		
4		2 DW002	4,5,6	de Wolden	MU06	muziek		
5		3 DW004	5	de Wolden	MU06	muziek		
6		4 DW005	1,2c	de Wolden	MU06	muziek		
7		5 DW005	6b	de Wolden	MU06	muziek		
8		6 DW006	1,2	de Wolden	MU06	muziek		
9		7 DW007	4	de Wolden	MU06	muziek		
10		8 DW008	5,6	de Wolden	MU06	muziek		
11		9 DW009	7,8	de Wolden	MU06	muziek		
12		10 DW011	3	de Wolden	MU06	muziek		
13		11 DW013	4	de Wolden	MU06	muziek		

Figuur 8: Elf workshops in twee dagen voor één docent.

Constraint 4: Tussen verschillende workshops van een groep moeten ten minste 30 dagen zitten.

Om te laten zien dat deze constraint werkt zijn er vier willekeurige workshops ingepland bij groep 5 van school HG008 in de gemeente Hoogeveen. Tussen elke workshop moeten ten minste 30 dagen zitten. Dus het kleinste interval waarop vier workshops gegeven kunnen worden loopt van dag 0 tot en met dag 90. Dag 0 was in deze situatie 3 september 2018 waardoor dag 90 op een zondag viel. Daardoor is ervoor gekozen om het programma als periode waarin gepland kan worden van dag 0 tot en met dag 91 mee te geven, zodat de situatie wel oplosbaar is. De resultaten hiervan zijn te vinden in Figuur 9. Er is goed te zien dat er minimaal 30 dagen tussen twee workshops zijn. Als bij dezelfde situatie de periode waarover gepland mag worden verkleind wordt met één dag, wordt de objective 0 en zijn er geen docenten en dagen meer gekoppeld aan de lesactiviteiten doordat de situatie onoplosbaar is geworden.

	A	B	C	D	E	F	G	H	I
1	objective	-200							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3		1 HG008	5	Hoogeveen	ME05	media	Ber	91	3-12-2018
4		2 HG008	5	Hoogeveen	TH06	theater	Eba	0	3-9-2018
5		3 HG008	5	Hoogeveen	MU16	muziek	Zuk	30	3-10-2018
6		4 HG008	5	Hoogeveen	ER01	erfgoed	Wie	60	2-11-2018

Figuur 9: Eén groep met vier verschillende workshops.

Constraint 5: Workshops uit een lessenserie moeten vier maal achter elkaar elke week op dezelfde dag worden ingepland met dezelfde docent.

Deze constraint is getest door lessenseries te plannen en te kijken of de docent bij een lessenserie hetzelfde is gebleven gedurende de vier lessen en elke les van de lessenserie zeven dagen later is dan de voorgaande les. De lessenseries ER05 en ME03 zijn gepland bij twee verschillende scholen. De resultaten staan in Figuur 10. Bij elke lessenserie staat vier keer dezelfde docent en de dag waarop de volgende les wordt gegeven is precies zeven dagen later.

	A	B	C	D	E	F	G	H	I
1	objective	-400							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	DW001	6	de Wolden	ER05	erfgoed	Wie	21	24-9-2018
4	2	DW001	6	de Wolden	ER05	erfgoed	Wie	28	1-10-2018
5	3	DW001	6	de Wolden	ER05	erfgoed	Wie	35	8-10-2018
6	4	DW001	6	de Wolden	ER05	erfgoed	Wie	42	15-10-2018
7	5	DW003	6,7,8	de Wolden	ME03	media	Ber	21	24-9-2018
8	6	DW003	6,7,8	de Wolden	ME03	media	Ber	28	1-10-2018
9	7	DW003	6,7,8	de Wolden	ME03	media	Ber	35	8-10-2018
10	8	DW003	6,7,8	de Wolden	ME03	media	Ber	42	15-10-2018

Figuur 10: Twee lessenseries.

Constraint 6: Een workshop kan niet gegeven worden aan een groep als de duur van de workshop groter is dan de intersectie van beschikbaarheid op een dag van een groep, docent en workshop.

Bij constraint 3 is laten zien dat de docent niet meer lessen kan geven dan waar hij/zij tijd voor heeft op een dag. Bij deze constraint wordt specifiek gekeken of op een dag een groep, docent en project wel beschikbaar zijn. Als ten minste één van deze factoren niet beschikbaar is, wordt de betreffende workshop voor de betreffende groep met de betreffende docent niet ingepland op die dag. Als alle drie de factoren wel aanwezig zijn op een dag gaat er gekeken worden naar de intersectie van de tijden van de groep met de docent. De workshop is de hele dag beschikbaar. Figuur 11 is het resultaat als workshop BE08 een afwezigheidsperiode heeft gekregen van 3 september tot en met 31 oktober. De gevraagde workshops BE08 worden precies na deze periode ingepland. Als een groep een deel van de ochtend en een deel van de middag aanwezig is, maar deze intervallen niet aaneengesloten zijn, en de docent de hele dag kan, worden er twee intervallen gemaakt. Zo wordt ook voldaan aan de hard constraint dat workshops niet onderbroken mogen worden door een pauze. Groepen 3, 4, 5 en 6 van de school met ID DW001 zijn op maandag aanwezig van 08:30 tot 12:00 uur en van 13:00 tot 15:00 uur. Ze hebben allemaal workshop DA13 aangevraagd. Bij deze workshop staat in de eerste prioriteitenlijst alleen docent Kri. Hij is de gehele maandag aanwezig. De workshop mag gegeven worden op deze maandag en heeft een duur van 60 minuten. De intersecties van de tijden van de groepen, docent Kri en de workshop bestaat in de ochtend uit 180 minuten en in de middag uit 120 minuten. Daardoor kan docent Kri precies vijf workshops DA13 geven op dezelfde dag. Als er meer workshops DA13 bijkomen, gaan die workshops naar een andere dag.

	A	B	C	D	E	F	G	H	I
1	objective	-100							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	DW001	3	de Wolden	BE08	beeldend	And	56	29-10-2018
4	2	DW001	5	de Wolden	BE08	beeldend	And	58	31-10-2018

Figuur 11: Workshop is afwezig van 3 september tot en met 28 oktober.

	A	B	C	D	E	F	G	H	I
1	objective	-250							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	DW001	3	de Wolden	DA13	dans	Kri	0	3-9-2018
4	2	DW001	4	de Wolden	DA13	dans	Kri	0	3-9-2018
5	3	DW001	5	de Wolden	DA13	dans	Kri	0	3-9-2018
6	4	DW001	6	de Wolden	DA13	dans	Kri	0	3-9-2018
7	5	DW001	7	de Wolden	DA13	dans	Kri	0	3-9-2018

Figuur 12: Docent Kri kan maximaal vijf workshops DA13 geven op één dag.

Constraint 7: Als een docent geen lesactiviteit gepland heeft op een dag dan is hij/zij niet aan het werk op die dag.

Constraint 8: Als een docent een lesactiviteit gepland heeft op een dag dan is hij/zij aan het werk op die dag.

Constraint 9: Er kunnen maximaal twee docenten media tegelijkertijd op een dag aan het werk zijn.

Constraint 7 en 8 zorgen ervoor dat constraint 9 werkt, waardoor deze drie constraint nauw met elkaar verbonden zijn. Constraint 7 en 8 zorgen er samen voor dat de variabele $w_{t,d}$ 1 wordt als de desbetreffende docent werkt op een dag en 0 wordt als de docent niet werkt. Daardoor kan er bij constraint 9 worden gekeken hoeveel docenten media er op een dag werken. Het effect van constraint 7 en 8 kunnen niet apart gecontroleerd worden aan de hand van het resulterende Excelbestand. Echter, als constraint 9 goed werkt betekent dit dat 7 en 8 ook op de juiste manier werken. Voor deze situatie zijn negen workshops van media ingepland die allemaal 90 minuten duren. Een docent kan maximaal, als de docent de hele dag werkt, drie workshops geven. Aangezien er negen workshops gepland moeten worden zijn er door deze constraint minstens twee dagen nodig. Dit is te zien in Figuur 13. Als er aangegeven wordt dat deze negen workshops op één dag gegeven moeten worden, wordt de objective nul en zijn er geen docenten ingeroosterd bij de lesactiviteiten. Workshop BE05 en BE09 zijn workshops waarbij vijf docenten in prioriteitenlijst 1 staan. Uit deze lijst zijn precies de docenten gekozen die het beste op die dag konden werken, omdat ze bij een andere gevraagde workshop de hoogste prioriteit hadden, waardoor er een optimale oplossing ontstaat.

	A	B	C	D	E	F	G	H	I
1	objective	-300							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	DW001	3	de Wolden	ME04	media	Opm	0	3-9-2018
4	2	DW005	1,2b	de Wolden	ME05	media	Hul	1	4-9-2018
5	3	DW005	6a	de Wolden	ME07	media	Opm	0	3-9-2018
6	4	DW007	5	de Wolden	ME08	media	Zie	1	4-9-2018
7	5	DW009	3,4	de Wolden	ME09	media	Opm	0	3-9-2018
8	6	DW013	4	de Wolden	ME11	media	Ber	0	3-9-2018

Figuur 13: Er werken maximaal twee docenten media op één dag.

Constraint 10: Een docent van de discipline beeldend kan maar drie lesactiviteiten beeldend geven op één dag.

De situatie waarmee deze constraint is getest heeft de volgende eigenschappen. Op verschillende scholen in de gemeente Meppel zijn in totaal vijf workshops van de discipline beeldend aangevraagd. Om de situatie goed te testen moeten al deze workshops gegeven worden op dinsdag 4

september 2018 en is er gekozen voor dezelfde workshop met ID BE01 aangezien de prioriteitlijst dan hetzelfde is. Alle docenten die deze workshop mogen geven zijn op 4 september de hele dag beschikbaar. In het rooster dat het programma als beste optie aangeeft, zie Figuur 14, is te zien dat de docent met de hoogste prioriteit (Bar) drie keer op die dag een workshop geeft. De docent zou meer workshops kunnen geven als deze constraint er niet was, aangezien de duur van de workshop voor het testen van deze constraint op 60 minuten is gezet en de docenten allemaal vijf uur kunnen werken. In Figuur 15 is de situatie te zien als constraint 10 er niet zou zijn. Er is te zien dat de objective dan wel een betere waarde heeft vergeleken met de situatie mét constraint 10. Dit komt doordat een docent met de hoogste prioriteit in de situatie met deze constraint maar drie workshops mag geven en dan wordt er doorgeschoven naar de docent in de volgende prioriteitenlijst. Docent Bar wordt dus in eerste instantie zoveel mogelijk ingepland, daarna wordt docent Lee erbij gehaald, aangezien deze docent in prioriteitenlijst 2 staat. De beste waarde die de objective had kunnen hebben zou $5 \cdot -50 = -250$ zijn. Door het toevoegen van constraint 10 is dit $3 \cdot -50 + 2 \cdot -20 = -190$ geworden. Deze constraint zorgt dus voor een slechtere objective, maar de constraint is noodzakelijk om een programma aan te leveren die voldoet aan de wensen van Scala.

	A	B	C	D	E	F	G	H	I
1	objective	-190							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	MP002	5	Meppel	BE01	beeldend	Lee	0	4-9-2018
4	2	MP003	1,2d	Meppel	BE01	beeldend	Bar	0	4-9-2018
5	3	MP004	1,2	Meppel	BE01	beeldend	Lee	0	4-9-2018
6	4	MP005	3,4,5c	Meppel	BE01	beeldend	Bar	0	4-9-2018
7	5	MP006	VSOPmid	Meppel	BE01	beeldend	Bar	0	4-9-2018

Figuur 14: Vijftien workshops van de discipline beeldend op één dag met constraint 10.

	A	B	C	D	E	F	G	H	I
1	objective	-250							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	MP002	5	Meppel	BE01	beeldend	Bar	0	4-9-2018
4	2	MP003	1,2d	Meppel	BE01	beeldend	Bar	0	4-9-2018
5	3	MP004	1,2	Meppel	BE01	beeldend	Bar	0	4-9-2018
6	4	MP005	3,4,5c	Meppel	BE01	beeldend	Bar	0	4-9-2018
7	5	MP006	VSOPmid	Meppel	BE01	beeldend	Bar	0	4-9-2018

Figuur 15: Vijftien workshops van de discipline beeldend op één dag zonder constraint 10.

Constraint 11: Een docent mag op één dag maar in één gemeente workshops geven.

De situatie die gekozen is om deze constraint te laten zien is als volgt. De workshop BE08 kan alleen maar gegeven worden door docent And. Deze workshop is vier keer ingedeeld bij verschillende scholen. Twee scholen staan in dezelfde gemeente. Zoals aan de resultaten is te zien, kunnen die twee workshops op dezelfde dag gegeven worden. De andere twee workshops zitten niet in dezelfde gemeente en daardoor wordt de docent genoodzaakt om deze workshops op een andere dag te geven. Als deze vier workshops gepland moeten worden in twee dagen wordt de objective nul doordat constraint 11 ervoor zorgt dat de situatie dan niet meer oplosbaar is.

	A	B	C	D	E	F	G	H	I
1	objective	-200							
2	Lesactiviteit	School ID	Groep	Gemeente School	Workshop ID	Discipline	Docent	Dag	
3	1	DW001	1,2	de Wolden	BE08	beeldend	And	0	3-9-2018
4	2	DW008	5,6	de Wolden	BE08	beeldend	And	0	3-9-2018
5	3	HG002	1	Hoogeveen	BE08	beeldend	And	2	5-9-2018
6	4	MP012	2b	Meppel	BE08	beeldend	And	1	4-9-2018

Figuur 16: *Docent mag op één dag niet in meerdere gemeentes lesgeven.*

6. DISCUSSIE EN AANBEVELINGEN

De resultaten in het vorige hoofdstuk suggereren over het algemeen dat alle constraints die bij dit model zijn meegenomen naar behoren werken. Er zijn zeker verbeteringen mogelijk die dit model uitbreiden en zorgen dat het programma uiteindelijk echt gebruikt kan worden door Scala. Op dit moment zijn niet alle punten die Scala graag zou zien meegenomen in het programma. Hieronder worden een aantal punten genoemd waardoor het programma verbeterd zou kunnen worden.

- Het kan nog voorkomen dat tussen de vierde les van een lessenserie en de volgende workshop van een groep slechts negen dagen zitten. Door een constraint toe te voegen die zegt dat 30 dagen na de vierde les van de lessenserie niks gepland mag worden, wordt in elke situatie toegepast dat er tussen geen enkele workshop minder dan 30 dagen mag zitten (met uitzondering van de lessenseries).
- Er gaat een grote verbetering komen als de tijd precies wordt meegenomen in het model. Dan komt het rooster dat het programma maakt al richting een rooster dat daadwerkelijk gebruikt kan worden. De docenten krijgen dan namelijk een rooster waar precies de tijden op staan dat ze een workshop moeten geven in plaats van alleen de dag, wat nu het geval is. Daarnaast kan er dan ook precies rekening gehouden worden met de reistijden. Er is nu geprobeerd rekening te houden met eventuele reistijden door een half uur tijd af te halen van elk dagdeel dat een docent werkt. Dit is echter een schatting die soms wel voor genoeg reistijd zorgt, maar ook vaak genoeg niet. Door een tijdsindeling te hebben kan ook hier nauwkeurig rekening mee gehouden worden.
- Door een precieze tijdsindeling te maken kan de soft constraint dat de lessen uit de lessenseries elke week precies op hetzelfde moment gegeven worden ook meegenomen worden. Deze soft constraint was echt een voorkeur van Scala wat niet per se in het programma verwerkt hoefde te worden, maar wat wel een mooie toevoeging zou zijn. Vandaar dat deze constraint nu nog niet is meegenomen, maar dit zou wel een verbetering zijn voor het programma vanuit Scala's oogpunt.
- Vaak gebeurt het in de planning dat niet vanaf de eerste dag wordt gepland, maar zomaar midden in het jaar, terwijl de eerste dag voor de betreffende groep, docent en workshop gewoon beschikbaar zijn. Dit is bijvoorbeeld te zien in Figuur 4 waar de meeste lesactiviteiten aan het begin worden gepland behalve de bovenste lesactiviteit die op dag 92 werd gepland en in Figuur 10 waar het programma pas begint met plannen op dag 21. Deze dagen zijn niet verkeerd, maar het is wel opvallend dat bepaalde lesactiviteiten niet aan het begin werden gepland terwijl dit bij de meeste andere lesactiviteiten wel gebeurt. Bij nader onderzoek leek dit alleen te gebeuren als er meerdere prioriteitenlijsten gevuld zijn van de betreffende workshop. Als alle prioriteitenlijsten leeg zijn of als alleen de eerste prioriteitenlijst gevuld

is, wordt wel vanaf de eerste dag gepland. De oorzaak hiervan is nog niet gevonden, maar doordat de dagen waarop de workshops gepland worden wel beschikbare dagen zijn voor de betrokken groepen, docenten en workshops lijkt het geen kwaad te doen. Toch is verder onderzoek hiernaar geen slecht idee, aangezien het misschien ook kan liggen aan een fout in de code.

- Bij datasets met veel lessenseries viel het op dat het plannen van de lessenseries niet altijd goed gaat. Soms komt het voor dat er verschillende docenten worden gekoppeld aan dezelfde lessenserie en soms kloppen de dagen niet. Zo is het een enkele keer voorgekomen dat het rooster doorgaat aan het einde van het schooljaar naar het begin van het schooljaar, waardoor er bijvoorbeeld een les van de lessenserie op dag 310 is gepland en de volgende les uit de lessenserie is op dag 3 gepland. Ook kwam het incidenteel voor dat de lessen uit de lessenseries niet precies zeven dagen later dan de vorige les waren gepland. De oorzaak hiervan is nog niet gevonden. Het wordt aangeraden om in de toekomst dit nog verder te onderzoeken, aangezien lessenseries een belangrijk onderdeel zijn van Scala die goed gepland moeten worden.
- De soft constraint dat docenten geen of zoveel mogelijk workshops moeten geven op een dag is in dit model nog helemaal niet meegenomen. Dit is een goed punt om mee te nemen in een volgend model, aangezien dit grote voorkeur had vanuit Scala.
- Bij veel docenten in het Excelbestand stond als opmerking dat ze maar een gedeelte van de ochtend of een gedeelte van de middag kunnen werken. Hier is toen een aanname gedaan dat als de docenten maar een gedeelte van het dagdeel kunnen werken, ze het hele dagdeel niet kunnen werken. Een verbetering hiervoor zou zijn als docenten ook precies aangeven welke tijden ze werken en deze werktijden vervolgens meegenomen worden in het programma. Dan kunnen deze docenten misschien toch net nog in de ochtend een workshop geven vanaf 10 uur in plaats van dat nu in het rooster geldt dat ze de hele ochtend niet kunnen.
- Door toe te voegen in welke gemeentes de docenten wonen kan er in het programma voor worden gezorgd dat alle docenten in de dichtsbijzijnde gemeentes werken om de reiskosten te verminderen. Hier waren nog geen gegevens van beschikbaar, zodoende is dit punt niet meegenomen in het model. Dit is een verbeterpunt voor een volgend model.
- Ook waren voorkeuren van de gemeentes waarin de docenten wilden werken verwaarloosd. In het Excelbestand van Scala waren deze gegevens al wel beschikbaar. Door hiervoor een constraint toe te voegen in het model, kan het programma ook verbeterd worden op deze voorkeuren. Als Scala dit wel wil, want dat was nog niet helemaal duidelijk.
- De workshops van de discipline multi zijn uit de gegevens gehaald aangezien hier geen docenten aan waren gekoppeld. Door in het Excelbestand wel docenten te koppelen aan deze workshops, kunnen ook deze workshops meegenomen worden in het roosteren. Dit wordt Scala aanbevolen om te doen als ze deze workshops toch willen plannen.
- Als dit programma echt gebruikt gaan worden door Scala om een rooster voor hen te maken, moet er een oplossing gevonden worden voor het probleem dat op dit moment het roosterprobleem niet op te lossen is op een gewone computer. De computer heeft hier onvoldoende geheugen voor de vele variabelen en gaat daardoor snel 'out-of-memory'. Het programma kan in dit opzicht verbeterd worden door bijvoorbeeld de zaterdag en zondag geheel niet mee te nemen in het programma en bij elke workshop niet te lopen over alle docenten, maar alleen over de docenten van die discipline. Dit zorgt voor minder variabelen,

maar waarschijnlijk zijn het er nog te veel voor de computer om dit aan te kunnen. Zodoende moet er nog goed gekeken worden naar hoe het programma verbeterd kan worden en of er andere mogelijkheden zijn om te zorgen dat een gewone computer genoeg geheugen heeft om het programma een rooster te laten produceren zodat Scala uiteindelijk het programma ook echt zelf kan gebruiken.

- Eventueel kan er in een volgend model worden toegevoegd dat er bij de workshops ook meerdere docenten kunnen staan in prioriteitenlijst 2 tot en met 8. Nu is dit alleen nog mogelijk in prioriteitenlijst 1. Dit is goed mogelijk om toe te voegen aan het programma, als Scala dit graag zou willen.

7. CONCLUSIE

Als bacheloropdracht is een programma ontwikkeld die op een exacte manier een rooster kan genereren voor Scala. Door onvoldoende geheugen van de computer en de ontbrekende gegevens van Scala kan er weinig gezegd worden over het echte roosterprobleem en over de oplosbaarheid van dit roosterprobleem. Het programma is wel draaiend gekregen met de constraints daarin verwerkt, die van tevoren gekozen waren. Nog niet alle voorwaarden die Scala wilde zien zaten in het model en daardoor ook niet in het programma. Er is dus nog zeker verbetering nodig voordat Scala dit programma daadwerkelijk kan gebruiken in plaats van een rooster te maken met de hand.

8. DANKWOORD

Na een intensieve periode is het einde van mijn bacheloropdracht in zicht. Met dit dankwoord leg ik de laatste hand aan mijn bacheloropdracht. Het was een periode waarin ik veel obstakels ben tegen gekomen, maar desondanks ook erg veel heb geleerd. Graag neem ik dit moment om een aantal personen te bedanken die mij de afgelopen periode veel hebben geholpen en hebben gesteund.

Allereerst wil ik graag mijn begeleider Gerhard Post bedanken voor alle hulp. Het was erg fijn dat we altijd bij u terecht konden en mede daardoor heb ik deze periode erg veel geleerd. De wekelijkse afspraken kwamen altijd precies op tijd; wanneer ik niet meer wist hoe het verder moest gaf u de hulp om weer verder te kunnen. U heeft mij de juiste handvatten aangereikt om de goede weg op te kunnen gaan.

Daarnaast wil ik Maarten Gal bedanken voor alle informatie over Scala die we zowel over de mail hebben gekregen als tijdens het moment aan het begin van de periode toen u speciaal hiervoor naar Enschede kwam.

Tenslotte wil ik mijn familie en vrienden bedanken voor de wijze raad en luisterend oor. In het speciaal Nico de Jongh, die dezelfde bacheloropdracht had als ik, waarmee ik altijd even kon sparren over onze problemen en bevindingen.

9. LITERATUURLIJST

- Dimopoulou M. & Miliotis P. (2001). Implementation of a university course and examination timetabling system. *European Journal of Operational Research*, Volume 130, Issue 1, pp. 202-213.
<https://www.sciencedirect.com/science/article/abs/pii/S0377221700000527>
- Havas, J., Olsson, A., Persson, J. & Schierscher, M. (2013). Modeling and optimization of university timetabling - a case study in integer programming. *Göteborgs universitets publikationer*.
<http://publications.lib.chalmers.se/records/fulltext/185558/185558.pdf>
- Ryan D. & Foster, B. (1981). An integer programming approach to scheduling. *Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling*, pp. 269-280.
<http://www2.imm.dtu.dk/courses/02735/ryanfoster.pdf>
- Schaerf, A. (1991). A Survey of Automated Timetabling. *Journal Artificial Intelligence Review*, Volume 13, Issue 2, pp. 87-127
<http://www.diegm.uniud.it/schaerf/SAS/articles/SurveyTimetabling.pdf>

10. APPENDIX

10.1. Punten waar tegenaan werd gelopen

De punten waar tegenaan werd gelopen lagen voornamelijk aan het aangeleverde excel bestand van Scala met alle gegevens. Hieronder staan de belangrijkste punten.

- Ons werd verteld dat docenten maar één discipline kunnen hebben. Echter, in het Excelbestand bleek dat veel docenten ingedeeld waren bij meerdere disciplines. Zo bleek zelfs dat de discipline erfgood, op één docent na, alleen maar bestond uit docenten die al een andere discipline hadden. Voor dit punt is nog niet duidelijk of de data verkeerd is of dat we op dit punt verkeerd zijn ingelicht.
- Voor het oplossen van het probleem is het noodzakelijk om van iedere docent de discipline te weten. Dit is bijvoorbeeld doordat als er achter een workshop geen docenten staan, alle docenten van de betreffende discipline ingedeeld kunnen worden. Nochtans stonden er bij veel docenten niet specifiek een discipline maar een beschrijving van wat diegene doet, zoals 'gitaar' en 'schrijver', maar ook minder zeggende uitspraken als 'AMV etc' en 'Meer vlog'. Bij sommige docenten stond zelfs helemaal geen discipline. De disciplines moeten dus duidelijker in het bestand gezet worden voordat een rooster gemaakt kan worden.
- Op veel plekken in het excel bestand stonden opmerkingen, zoals dat scholen volgend jaar gaan fuseren en daarom niet meer bestaan of dat een bepaalde groep op maandagmiddag afwezig is. We hebben deze opmerkingen moeten weghalen, omdat anders het programma om het rooster te maken niet werkt. Het was te tijdrovend om zelf alle opmerkingen te verwerken in de gegevens. We gaan ervan uit dat Scala deze punten zelf verwerkt in de gegevens zodat deze punten uiteindelijk wel worden meegenomen.
- Het was niet duidelijk wat een geel hokje precies inhield bij de beschikbaarheid van de docenten. Daardoor is hiervoor een aanname gedaan (zie paragraaf 3.2).
- We wilden graag rekening houden met de gemeente waar elke docent woont zodat de reistijd geminimaliseerd kon worden. Aangezien we hiervan geen gegevens binnen hebben gekregen, hebben we dit punt verwaarloosd in het model.
- Achter alle workshops staan docenten die die betreffende workshop geven (of als er niets staat: alle docenten van die discipline). Hierbij zijn meestal de achternamen van de docenten gebruikt, maar bij nader onderzoek ook soms de voornaam. Een enkele keer kwam het zelfs voor dat de docent die bij een workshop stond niet voorkwam in de lijst met docenten. Doordat van deze docenten geen gegevens beschikbaar waren, zijn deze docenten verwijderd bij de workshops. Daarnaast stonden er heel veel docenten in de docentenlijst die niet ingedeeld waren bij een workshop. Doordat de disciplines niet goed genoemd zijn bij de docenten was het niet duidelijk wat deze docenten wel en niet kunnen geven. Tevens kwam het voor dat er een docent werd genoemd achter een workshop, maar dat er in de lijst met docenten meerdere keren deze naam voorkwam waardoor het niet duidelijk was over welke docent het ging. Dit was belangrijk om te weten, onder andere om te weten wanneer een docent beschikbaar is. Het is dus aan Scala om te zorgen dat dit er allemaal netjes inkomt en er geen overlap is.
- Een enkele keer kwam het voor dat prioriteiten van docenten niet goed gedefinieerd waren. Zo kwam bij één workshop de prioriteiten 1, 2, 2 en 4 voor en bij een andere workshop 1, 2,

3 en dan drie docenten zonder prioriteit. Of dit typfouten waren of dat het wel goed stond, was niet duidelijk.

- Een aantal klassen hadden geen gegevens over hun schooltijden. Deze groepen zijn verwijderd uit de gegevens aangezien ze volgens het programma nooit tijd hebben om een workshop te krijgen en daardoor voor een onoplosbaar roosterprobleem zorgen.
- Een punt dat onduidelijk bleef vanuit Scala was of ze docenten wilden clusteren op één dag voor één klas bij verschillende scholen of dat ze docenten wilden clusteren op één dag bij dezelfde school. Aangezien Scala niet goed wist wat ze liever wilde, is dit punt volledig buiten beschouwing gelaten.

10.2. Layout data Scala

Om het programma goed te laten werken is het noodzaak dat de data goed aangeleverd wordt. Hieronder wordt uitgelegd hoe het Excelbestand er uit moet komen te zien. Er zijn vier tabbladen. Bij elk tabblad staat wat de de titel van het tabblad is, wat de titel van elke kolom is en zowel wat er in die kolom te vinden is als hoe het ingevuld moet worden.

Tabblad 1: School

1. ID: Hier komt de unieke ID van de school te staan. Deze bestaat uit twee letters die de gemeente aanduiden met daarna een getal van drie cijfers.
2. Naam: De naam van de school staat in deze kolom.
3. woonplaats: Hier komt de plaats te staan waar de school in staat. Deze gegevens worden in het huidige model niet gebruikt.
4. Gemeente: Deze kolom geeft de gemeente aan waar de school zich bevindt.
5. Speciaal: Hier kan aangegeven worden of het een school is met speciaal onderwijs (ja) of regulier onderwijs (nee). Deze gegevens worden in het huidige model niet gebruikt. Om toch mee te nemen dat er scholen zijn met speciaal onderwijs is het de bedoeling dat er speciale workshops worden aangemaakt in de lijst met workshops voor speciaal onderwijs. Achter deze workshops moet aangegeven worden welke docenten deze workshops mogen geven. Dit zullen in dit geval alleen docenten zijn die bevoegd zijn in lesgeven op het speciaal onderwijs. Als de groepen in het speciaal onderwijs workshops uitkiezen is het van belang dat hier de workshops worden ingevuld die bestemd zijn voor het speciaal onderwijs. Zo krijgen de leerlingen op het speciaal onderwijs alleen les van docenten die hiervoor bevoegd zijn.
6. Groepen (aantal klassen): Hoeveel klassen een school heeft wordt hier aangegeven. Deze gegevens worden in het huidige model niet gebruikt.
7. Startdatum: Hier kan de startdatum van het schooljaar ingevuld worden. Deze gegevens worden echter in het huidige model niet gebruikt. Het is de bedoeling dat het begin vanaf dat er gepland mag worden ingevuld wordt in het programma bij "Startdatum", zie Figuur 1.
8. Einddatum: Hier kan de einddatum van het schooljaar ingevuld worden. Deze gegevens worden echter in het huidige model niet gebruikt. Het is de bedoeling dat tot wanneer er gepland mag worden ingevuld wordt in het programma bij "Einddatum", zie Figuur 1.

9. Maandag: Hier kan "nee" worden ingevuld als alle klassen van die school die dag niet beschikbaar zijn op maandag.
10. Dinsdag: Hier kan "nee" worden ingevuld als alle klassen van die school die dag niet beschikbaar zijn op dinsdag.
11. Woensdag: Hier kan "nee" worden ingevuld als alle klassen van die school die dag niet beschikbaar zijn op woensdag.
12. Donderdag: Hier kan "nee" worden ingevuld als alle klassen van die school die dag niet beschikbaar zijn op donderdag.
13. Vrijdag: Hier kan "nee" worden ingevuld als alle klassen van die school die dag niet beschikbaar zijn op vrijdag.
14. Afwezigheidsperiode: Hier kan door "dd-mm-jjjj t/m dd-mm-jjjj" in te vullen aangegeven worden als een school een periode niet aanwezig is, bijvoorbeeld tijdens een vakantie. Meerdere kolommen zijn hier mogelijk.

Tabblad 2: Groepen

1. School: De ID van de school waar de groep, die in de volgende kolom staat, bijhoort.
2. Groep: Van de school in de eerste kolom worden hier alle groepen weergegeven. Voor duidelijkheid in het rooster is het belangrijk dat bij een school niet meerdere malen dezelfde groepen voorkomen. Dit is op te lossen door bijvoorbeeld groep 1a en 1b te maken.
3. Workshop (4 maal): In deze kolommen kunnen maximaal vier workshops opgegeven worden voor de betreffende groep. Belangrijk is dat hier niet de naam van de workshop komt, maar de ID van de workshop.
4. Maandagtijden: Door "hh:mm tot hh:mm en hh:mm tot hh:mm" of "hh:mm tot hh:mm" kan hier aangegeven worden wat de tijden zijn waarop de betreffende groep die dag aanwezig is op maandag.
5. Dinsdagtijden: Door "hh:mm tot hh:mm en hh:mm tot hh:mm" of "hh:mm tot hh:mm" kan hier aangegeven worden wat de tijden zijn waarop de betreffende groep die dag aanwezig is op dinsdag.
6. Woensdagtijden: Door "hh:mm tot hh:mm en hh:mm tot hh:mm" of "hh:mm tot hh:mm" kan hier aangegeven worden wat de tijden zijn waarop de betreffende groep die dag aanwezig is op woensdag.
7. Donderdagtijden: Door "hh:mm tot hh:mm en hh:mm tot hh:mm" of "hh:mm tot hh:mm" kan hier aangegeven worden wat de tijden zijn waarop de betreffende groep die dag aanwezig is op donderdag.
8. Vrijdagtijden: Door "hh:mm tot hh:mm en hh:mm tot hh:mm" of "hh:mm tot hh:mm" kan hier aangegeven worden wat de tijden zijn waarop de betreffende groep die dag aanwezig is op vrijdag.
9. Afwezigheidsperiode: Hier kan door "dd-mm-jjjj t/m dd-mm-jjjj" in te vullen aangegeven worden als een groep een periode niet aanwezig is. Meerdere kolommen zijn hier mogelijk.

Tabblad 3: Docenten

1. ID: In deze kolom is een ID gegeven aan elke docent. Gebruikelijk is het om de achternaam te nemen. Belangrijk is dat verschillende docenten niet dezelfde ID krijgen.
2. naam: De achternamen van de docenten staan in deze kolom.
3. tussenv: Hier kunnen eventuele tussenvoegsels genoemd worden.
4. roepnaam: De roepnamen van de docenten kunnen in deze kolom ingevuld worden. Zowel de gegevens van de achternaam, het tussenvoegsel als de roepnaam worden in het huidige model niet gebruikt.
5. discipline: Deze kolom is bestemd voor de noemen van de discipline van elke docent. De mogelijkheden zijn hier: "Beeldend", "Dans", "Erfgoed", "Media", "Muziek" of "Theater".
6. ma ochtend: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op maandagochtend niet aanwezig is.
7. ma middag: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op maandagmiddag niet aanwezig is.
8. di ochtend: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op dinsdagochtend niet aanwezig is.
9. di middag: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op dinsdagmiddag niet aanwezig is.
10. wo ochtend: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op woensdagochtend niet aanwezig is.
11. wo middag: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op woensdagmiddag niet aanwezig is.
12. do ochtend: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op donderdagochtend niet aanwezig is.
13. do middag: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op donderdagmiddag niet aanwezig is.
14. vr ochtend: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op vrijdagochtend niet aanwezig is.
15. vr middag: Als hier "Nee" ingevuld is, betekent dit dat de betreffende docent op vrijdagmiddag niet aanwezig is.
16. Afwezigheidsperiode: Hier kan door "dd-mm-jjjj t/m dd-mm-jjjj" in te vullen aangegeven worden als een docent een periode niet aanwezig is. Meerdere kolommen zijn hier mogelijk.

Tabblad 4: Projecten

1. projectcode: In deze kolom is een ID gegeven aan een project. Deze bestaat uit de eerste twee letters van de discipline met daarachter een getal van twee cijfers.
2. naam_project: Hier kan de naam van het project gegeven worden. Deze gegevens worden in het huidige model niet gebruikt.
3. discipline: Deze kolom is bestemd voor het noemen van de discipline van elke workshop.

4. aantal keer: In deze kolom is aangegeven hoe vaak een workshop gegeven moet worden. Als er "1" staat is het een eenmale workshop, bij "4" is er sprake van een lessenserie.
5. duur: De duur van elke workshop in minuten is weergegeven in deze kolom.
6. Docenten met prioriteit1: Alle docenten die bij de betreffende workshop de hoogste prioriteit hebben staan in deze kolom. Als een plaats leeg is betekent dat alle docenten van die discipline de workshop mogen geven. Er kunnen ook meerdere docenten in deze kolom op een plek staan, deze docenten moeten dan door een komma worden onderscheiden. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
7. Docenten met prioriteit2: De docenten van de betreffende workshop met prioriteit 2 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
8. Docenten met prioriteit3: De docenten van de betreffende workshop met prioriteit 3 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
9. Docenten met prioriteit4: De docenten van de betreffende workshop met prioriteit 4 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
10. Docenten met prioriteit5: De docenten van de betreffende workshop met prioriteit 5 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
11. Docenten met prioriteit6: De docenten van de betreffende workshop met prioriteit 6 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
12. Docenten met prioriteit7: De docenten van de betreffende workshop met prioriteit 7 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
13. Docenten met prioriteit8: De docenten van de betreffende workshop met prioriteit 8 staan in deze kolom. Belangrijk is dat van de docenten die hier geplaatst worden, hun ID wordt genoemd.
14. Afwezigheidsperiode: Hier kan door "dd-mm-jjjj t/m dd-mm-jjjj" in te vullen aangegeven worden als een project een periode niet gegeven mag worden. Meerdere kolommen zijn hier mogelijk.