

# Het ritme van Benchmark® Ultra

Algoritme ontwikkeling voor het optimaal vullen van immunokleuring machines bij de afdeling Pathologie

**Peter Wijbenga**

UMCG, Pathologie en Medische Biologie  
NHL, Bedrijfswiskunde

Groningen, 21 juni 2010

Studentenbureau UMCG

Universitair Medisch Centrum Groningen





## **Het ritme van Benchmark® Ultra**

Algoritme ontwikkeling voor het optimaal vullen van immunokleuring machines bij de afdeling Pathologie

Groningen, 21 juni 2010

Auteur  
Studentnummer

Afstudeerrapport in het kader van

Opdrachtgever

Begeleiders UMCG

Begeleiders onderwijsinstelling

Peter Wijbenga  
90811

Bedrijfswiskunde  
Educatie en Communicatie, Exacte Vakken  
NHL Hogeschool te Leeuwarden

O.H.J. Wieringa, hoofd Laboratorium Histologie  
Pathologie en Medische Biologie, UMCG

ir. A.P. Goudswaard  
hoofd Zorglogistiek & Innovatie

K.J. Wiering, M. Litjens  
Instituut Educatie en Communicatie, Exacte  
Vakken, Noordelijke Hogeschool Leeuwarden

ISBN 978-90-8827-069-7

NUR 981

Trefw Algoritme, Bedrijfskunde, Pathologie, Benchmark® Ultra

Omslag: Wenckebach Instituut, Universitair Medisch Centrum Groningen

© 2011 Studentenbureau UMCG Publicaties Groningen, Nederland.

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen, of enige andere manier, zonder voorafgaande toestemming van de uitgever.

Voor zover het maken van kopieën uit deze uitgave is toegestaan op grond van artikel 16B Auteurswet 1912 j° het Besluit van 20 juni 1974, St.b. 351, zoals gewijzigd in Besluit van 23 augustus 1985, St.b. 471 en artikel 17 Auteurswet 1912, dient men de daarvoor wettelijk verschuldigde vergoedingen te voldoen aan de Stichting Reprorecht. Voor het overnemen van gedeelte(n) uit deze uitgave in bloemlezingen, readers en andere compilatiewerken (artikel 16 Auteurswet 1912) dient men zich tot de uitgever te wenden.

## Voorwoord

De afdeling Pathologie op het Universitair Medisch Centrum te Groningen heeft kennis gegeven van hun probleem betreffende het zo optimaal mogelijk indelen van immunokleuring machines. Dit probleem heb ik, Peter Wijbenga, met beide handen aangepakt voor het afronden van mijn opleiding Bedrijfskunde aan de NHL Hogeschool te Leeuwarden. Over de aanpak van het probleem en de oplossingen en resultaten hiervan kunt u meer lezen in dit afstudeerrapport.

Zonder het betreffende probleem en de vraag om hier een oplossing voor te vinden, zou ik niet een dergelijk interessant afstudeerrapport hebben kunnen schrijven. Hiervoor wil ik Dhr. O.H.J. Wieringa en de medewerkers van de afdeling Pathologie en Laboratoriumgeneeskunde die, op welke wijze dan ook, hebben bijgedragen aan de vordering van mijn afstudeerrapport hartelijk bedanken. Ik wil graag dhr. J. Pols en mevr. A. Muurman van het Studentenbureau UMCG bedanken voor de begeleiding en de mogelijkheid die ik heb gekregen om mijn afstudeerrapport op het UMCG te mogen schrijven.

Tevens bedank ik voor de begeleiding Ir. A.P. Goudswaard vanuit het UMCG en Dhr. K.J. Wieringa vanuit de NHL Hogeschool.

Tot slot wil ik de overige medewerkers van het UMCG, medestudenten en/of vrienden die voor een aangename afstudeerperiode hebben gezorgd vriendelijk bedanken.

Peter Wijbenga  
Groningen, 25 mei 2011



# INHOUDSOPGAVE

<b>SAMENVATTING.....</b>	<b>1</b>
<b>SUMMARY .....</b>	<b>2</b>
<b>LIJST AFKORTINGEN EN BEGRIPPEN .....</b>	<b>3</b>
<b>1 INLEIDING .....</b>	<b>5</b>
1.1 INLEIDING.....	5
1.2 AANLEIDING.....	5
1.3 MOTIVATIE .....	5
1.4 PROBLEEMSTELLING.....	5
1.5 OPBOUW VAN HET RAPPORT.....	6
<b>2 ACHTERGRONDINFORMATIE .....</b>	<b>7</b>
2.1 ORGANISATIE.....	7
2.2 BENCHMARK@ ULTRA.....	8
<b>3 PROBLEEMSTELLING EN DOELSTELLING .....</b>	<b>11</b>
3.1 AANLEIDING PROBLEEMSTELLING.....	11
3.2 GEWENSTE SITUATIE .....	11
3.3 VOORWAARDEN.....	11
3.3 VRAAGSTELLING .....	12
3.4 DOELSTELLING.....	12
<b>4 OPZET EN UITVOERING.....</b>	<b>13</b>
4.1 AANLEIDING EN DOEL VAN HET ONDERZOEK .....	13
4.2 INHOUDELIJKE ORIËNTATIE .....	13
4.3 AANPAK VAN HET ONDERZOEK.....	13
<b>5 RESULTATEN.....</b>	<b>15</b>
5.1 ONTWIKKELING VAN HET ALGORITME .....	15
5.2 HET DEFINITIEVE ALGORITME.....	17
5.3 ALGORITME “IMMUNOCALCULATOR” .....	18
5.4 HET ALGORITME TESTEN .....	21
<b>6 CONCLUSIES .....</b>	<b>23</b>
<b>7 AANBEVELINGEN.....</b>	<b>25</b>
<b>8 EVALUATIE .....</b>	<b>27</b>

8.1 PROCES-EVALUATIE .....	27
8.2 PRODUCTEVALUATIE .....	28
<b>LITERATUUR .....</b>	<b>29</b>
BIJLAGE 1 LIJST VAN IMMUNOKLEURINGEN.....	31
BIJLAGE 2 ELKE MACHINE HEEFT EEN VASTE KLEURING.....	33
BIJLAGE 3 AANPASSEN DISPENSERPLEKKEN ONDER VOORWAARDE .....	34
BIJLAGE 4 SCHEMA ROOSTERPROBLEEM.....	35
BIJLAGE 5 THE BENCHMARK® ULTRA.....	36



## Samenvatting

Als afstudeeropdracht voor de opleiding Bedrijfswisskunde aan de NHL Hogeschool te Leeuwarden heb ik geprobeerd om een waardige oplossing te vinden voor een probleem dat ontstond op de afdeling Pathologie van het Universitair Medisch Centrum te Groningen. Vlak voordat ik deze opdracht kreeg had deze afdeling vier zo genaamde kleuringmachines aangeschaft, genaamd Benchmark® Ultra. Hiermee hebben ze het aanbrengen van immunokleuringen op weefsels geautomatiseerd. Het probleem van deze machines is dat ze ingedeeld moeten worden met de hand en aangezien de ruimte voor dispensers en coupes beperkt is, vormt zich hier de aanleiding van het probleem. Een dispenser is een buisje met reagentia, een benodigde stof met antilichamen om een kleuring te verrichten en een coupe is een glaasje met daarop een plakje weefsel (2 à 3 micrometer dun), gemaakt voor histologisch onderzoek.

Elke machine biedt plaats aan 30 coupes en 35 dispensers. Aan standaard reagentia zijn 8 van de 35 dispenserplekken nodig. Daarnaast vergt een aantal kleuringen meer dan 1 positie. Afhankelijk van deze hoeveelheid blijven er posities over om dispensers te plaatsen en verschillende kleuringen gelijktijdig te realiseren.

Het is mij gelukt een goed werkend algoritme te schrijven, rekening houdend met alle voorwaarden die aan het indelen van de machines gebonden zijn. Desondanks heb ik niet kunnen voldoen aan alle wensen van de afdeling Pathologie.

Als men mijn algoritme in een computerapplicatie opneemt, moeten de Pathologen namelijk zelf het aantal coupes van iedere kleur tellen. Dit moeten ze dan in het programma typen en vervolgens kunnen ze door een druk op de knop de indeling aflezen van het beeldscherm. Dat tellen is best veel werk. Een oplossing hiervoor kan het scannen van de streepjescodes op de coupes kunnen zijn. Als de Pathologen met een externe scanner de coupes kunnen inlezen in het programma, bespaard dit veel werk. Er valt nog meer tijd te besparen door de

coupes niet te scannen, maar al vanaf de eerste aanvraag direct door te laten sturen naar de applicatie. Hiervoor zal de applicatie moeten onthouden hoeveel coupes van elke kleur doorgegeven zijn. Dit vergt echter nog wel veel werk, maar is voor de nabije toekomst zeker realiseerbaar!

De Pathologen wouden graag zoveel mogelijk kleuren op één bepaalde machine houden, om zo te voorkomen dat ze over en weer moeten slepen met de dispensers van deze kleuren. Dit vergemakkelijkt ook het doorladen van de coupes, waar in de toekomst waarschijnlijk meer gebruik van wordt gemaakt. Ik ben niet nagegaan op welke machine de kleuren die regelmatig voorkomen het beste ingedeeld kunnen worden, omdat er niet genoeg relevante data aanwezig was om dit te kunnen onderzoeken.

Aangezien er geen bestaande oplossingen voor een vergelijkbaar indelingsprobleem zijn, heb ik het gehele algoritme zelf moeten schrijven. Door slim na te denken en logisch te redeneren heb ik een zeer degelijk algoritme gemaakt. Het algoritme is getest met gebruikelijke/werkelijke waarden en laat als resultaat een prima indeling van de machines zien!

Het definitieve algoritme is zowel in verhaallijn als in algoritmetaal geschreven. Deze zijn te vinden in Hoofdstuk 5. Het algoritme in verhaallijn is ter ondersteuning van het echte algoritme. Het echte algoritme schreef ik in Pseudotaal; formuleringen in de moedertaal die men gebruikt om een algoritme te schrijven en een taal die de computer kan begrijpen.

Ik beveel de afdeling Pathologie aan om een programmeur mijn algoritme te laten verwerken in een computerapplicatie. Als het programma werkt dan raad ik aan eerst te proberen via een scanner de coupes in te lezen in het programma. Dit zal de afdeling veel tijd besparen. Als dit lukt dan is er aan de meeste wensen voldaan en kan men kijken naar een vaste indeling voor iedere machine. In de verre toekomst is het misschien zelfs mogelijk een goede computerapplicatie samen te voegen met de gebruikersinterface van de machines.

## Summary

As task for my graduation at the NHL University Leeuwarden, I have tried to find a worthy solution for a problem that arose at the Pathology department from the Universal Medical Centre Groningen. This department had purchased for colouring machines called Benchmark® Ultra, just before I was assigned to this task. These colouring machines can automatically do a part of immunohistological research by bringing a colour to some tissues.

The problem of these machines is that they have to be classified by hand. A problem occurs because of the limited space for dispensers and coupes. A dispenser is a small tube with reagents, the necessarily substance with antibodies to do the colouring. A coupe is a piece of glass containing a sliced tissue (2 to 3 micrometers thin), made for histological research.

Each machine has place for 30 coupes and 35 dispensers. Standard reagents take 8 of these 35 dispenser positions. Besides this, some colours take more than one position. It depends on these positions how many places remain to place other dispensers to realize more colours at the same time.

I succeeded to write a well working algorithm, taking all requirements for the classification into account. Anyhow I could not fulfil all wishes of the Pathology department. The pathologists have to count the coupes of every colour for themselves if my algorithm is included in a computer application, because the application has to know of each colour how many coupes are presented. They can read the classification from the display by pressing a button, after they have typed the number of coupes per colour in the application. Counting these coupes takes a lot of time. A solution therefore is to scan each barcode from every coupe. It would save a lot of time if the pathologists could use an external scanner to read the number of coupes per colour into the application. They can save even more time by sending the number of coupes to the application from the first request for colouring. For this the application will have to remember how many coupes of each colour are

sent. This requires still a lot of work, but is certainly feasible for the near future!

The pathologists would like to have as many as colours on one given machine, to avoid dragging the dispensers over and over again. This will also ease loading the coupes after each other, what is more likely for the future. I have not calculated on which machine the colours that appear frequently can be classified best, because there was not enough relevant data presented.

Since there are no existing solutions for a similar classification problem, I had to write the whole algorithm by myself. With smart thinking and logical reasoning, I have made a well working algorithm. The algorithm was tested with normal / actual values and the result shows a good classification of the machines!

The final algorithm is both written in storyline and in algorithm language. You can find this algorithm in Chapter 5. The algorithm in storyline is written to support the real algorithm. The real algorithm is written in a sort of intermediate language; a language that is generated from programming source code, but that cannot be directly executed by the computer.

I can recommend the Pathology department to let my algorithm be processed in a computer application by a programmer. If this works, then I recommend to try using a scanner to read the coupes into the program, because this will save the employees a lot of time. This will mostly satisfy the needs of the department. Next will be to calculate a (more) fixed classification for each machine. In the future, it may be even possible to merge a good computer application with the user interface of the machines! .

## Lijst afkortingen en begrippen

<b>Algoritme</b>	Nauwkeurig gedefinieerde regels of procedure om een gesteld resultaat te verkrijgen uit een eindig aantal bewerkingen. Goede algoritmes vormen vaak de basis van een efficiënt computerprogramma.
<b>Antilichaam</b>	Molecuul dat aan antigenen hecht om ze herkenbaar te maken.
<b>Coupes</b>	Een glaasje met daarop een dun plakje weefsel (2 à 3 micrometer) voor histologisch onderzoek.
<b>Dispensers</b>	Een buisje/houder met reagentia.
<b>Immuunhistochemie</b>	Bij immuunhistochemie wordt gebruik gemaakt van antilichamen om structuren in weefsel te markeren.
<b>Immunokleuring</b>	Bij een immunokleuring worden antigenen (eiwitten) in weefsel aangetoond met behulp van een antilichaam.
<b>Kleursoort</b>	Er zijn verschillende "kleursoorten": Ultraview, I-view, Eber, Rood en Dual-Sish. Hierbij moet vermeld worden dat Eber en Dual-Sish eigenlijk geen kleursoorten zijn, maar voor de duidelijkheid in dit rapport wel zo genoemd worden. Deze twee zijn namelijk gewoon de naam van de kleur.
<b>Pathologie</b>	De afdeling pathologie stelt aan de hand van microscopisch onderzoek van afgenomen lichaamsmateriaal vast om welke ziekte of aandoening het gaat.
<b>PMB</b>	Pathologie en Medische Biologie
<b>Pseudotaal</b>	Formuleringen in de moedertaal die gebruikt worden om een algoritme te schrijven.
<b>Reagentia</b>	Benodigde stof met antilichamen om een kleuring te doen.
<b>UMCG</b>	Universitair Medisch Centrum Groningen



## 1 Inleiding

### 1.1 Inleiding

Het ontwikkelen van een algoritme voor het optimaal vullen van immunokleuring machines. Van de opdracht met dit doel wordt in dit rapport het proces dat ik belopen heb van begin tot eind beschreven.

### 1.2 Aanleiding

Als opdracht voor mijn afstuderen leek het probleem, dat vlak voor mijn afstudeerstage ontstond op de afdeling Pathologie op het Universitair Medisch Centrum te Groningen (UMCG), mij erg interessant. De afdeling Pathologie heeft het aanbrengen van immunokleuringen geautomatiseerd door vier Benchmark® Ultra machines aan te schaffen. Deze machines verrichten nu dagelijks de kleuringen die voorheen met de hand gedaan moesten worden. Het probleem van deze machines is dat ze dagelijks ingedeeld moeten worden en aangezien de ruimte voor dispensers en glaasjes met weefsel (coupes) beperkt is, vormt zicht hier de aanleiding van het probleem. Hoe vullen we de machines op een manier dat er zoveel mogelijk kleuringen plaats kunnen vinden in een enkele draai?

### 1.3 Motivatie

Vanaf het begin dat ik het probleem onder ogen kreeg, was ik erg gemotiveerd om hier een oplossing voor te vinden. Juist omdat ik totaal onbekend ben op het gebied van Pathologie, ben ik mij hier zeer voor gaan interesseren. Door de vele voorwaarden die het probleem nog gecompliceerder maken vind ik het alsmaar interessanter worden. Zonder uitdaging vind ik geen plezier en motivatie om een oplossing te zoeken. De uitdaging heb ik in dit

probleem zeker gevonden en ben daardoor tot een goed eindresultaat gekomen.

Tevens vind ik het erg interessant om op onbekend terrein te opereren, omdat ik hierdoor veel samen heb moeten werken met Pathologen om zo de nodige informatie uit hen te halen. Soms was belangrijke informatie voor de Pathologen zo logisch, dat ik het niet prijs kreeg. Juist dit maakt het onbekende terrein voor mij interessant. Daarnaast vind ik het leuk om Pathologen iets bij te brengen op het gebied van Wiskunde. Veel mensen staan er niet bij stil hoeveel er met dit vak mogelijk is. Het goed duidelijk maken van mijn plan van aanpak was op zichzelf al een uitdaging.

### 1.4 Probleemstelling

Op de afdeling Pathologie zijn vier machines beschikbaar voor de immunokleuringen, genaamd Benchmark® Ultra. Bij immunokleuringen worden antigenen (eiwitten) in weefsel aangetoond met behulp van een antilichaam.

Het is aan te raden, maar niet noodzakelijk, om voor het lezen van de rest van het rapport een demo te bekijken van deze machines. Deze demo is te vinden op: <http://www.ventanamed.com/benchmarkultra/demo.php>

Elke machine biedt plaats aan 30 coupes. Coupes zijn glaasjes met weefsel (2 à 3 micrometer dun) die gekleurd moeten worden. Per machine zijn 35 posities beschikbaar om dispensers met reagentia (benodigde stof om een kleuring te doen) te plaatsen. Het kleuringassortiment bestaat op dit moment uit ongeveer 160 verschillende kleuringen. Sommige daarvan zijn dagelijkse kost, andere worden sporadisch aangevraagd. Aan standaard reagentia zijn minimaal 8 van de 35 posities nodig. Een aantal kleuringen vergt meer posities. Afhankelijk van deze hoeveelheid blijven er posities over om antilichamen te

plaatsen en verschillende kleuringen gelijktijdig te realiseren. Ook de resthoeveelheid van de basisreagentia kan het aantal posities nadelig beïnvloeden. Dagelijks worden deze machines ingedeeld door middel van handwerk. Het is de bedoeling hier een algoritme voor te schrijven, zodat het indelen van deze machines beter verloopt en minder tijd in beslag neemt.

## 1.5 Opbouw van het rapport

Dit rapport is als volgt opgebouwd:

Ik begin met de achtergrondinformatie. Zo krijgt u een duidelijk beeld van de omgeving waarin ik gewerkt heb en waarin het probleem is ontstaan.

Vervolgens zal ik in Hoofdstuk 3 de probleemstelling en doelstelling uitwerken. Hier wordt dus de kern van het probleem beschreven met alle randvoorwaarden waar ik mee te maken had.

In het vierde hoofdstuk zal ik de manier waarop in te werken gegaan en mijn gedachtegang hierbij proberen te omschrijven. De inhoudelijke oriëntatie is in paragraaf 4.2 beschreven.

In Hoofdstuk 5 geef ik een beeld van het ontstaan van mijn definitieve algoritme. Uiteraard zijn er wat probeersels aan mijn definitieve algoritme voorafgegaan. Het definitieve algoritme en het testen hiervan zijn ook in dit hoofdstuk te vinden.

Hoofdstuk 6, 7 en 8 bevatten respectievelijk de conclusies, aanbevelingen en de evaluatie.

De schriftelijke en digitale bronvermelding zal volgen en als laatst volgen de bijlagen waar in dit document naar verwezen kan worden. Er zijn ook bijlagen waar niet naar verwezen is, maar die ik wel belangrijk genoeg gevonden heb om op te nemen in de bijlagen.

## 2 Achtergrondinformatie

### 2.1 Organisatie

#### *Het UMCG*

Het UMC Groningen is het enige universitair medisch centrum in het noorden van het land. Het heeft alle medische specialismen en subspecialismen in huis en is eindpunt van verwijzing voor ongeveer drie miljoen Nederlanders. In het ziekenhuis is plaats voor ruim 1300 patiënten. Er werken zo'n 9000 mensen; jaarlijks groeit de werkgelegenheid in het UMCG met ongeveer 150 arbeidsplaatsen. Hiermee is het UMCG een van de grootste werkgevers in het noorden van het land. Het UMCG is een belangrijk kenniscentrum. In het centrum wordt voortdurend gewerkt aan verbetering van de organisatie van zorg en nieuwe zorgvormen en behandelmethoden. Ook doet het UMCG veel medisch wetenschappelijk onderzoek.

Het UMCG heeft gekozen voor een kikker als logo omdat het te typeren is als: vertrouwd, eigenzinnig en academisch. Een eigenzinnige keuze die past bij het UMCG. Een kikker is een alledaags dier dat zich prettig voelt in een schone en rustige omgeving waar het goed toeven is. Het symboliseert leven, gezondheid, vertrouwen en kwaliteit. Het academische komt tot uiting in de schematische weergave van het achterlijf. De stippel- en streepatronen zijn geïnspireerd op moderne medische scantechnieken.

#### *De afdeling Pathologie en Laboratoriumgeneeskunde*

De Pathologie is in het UMCG ondergebracht bij de afdeling Pathologie en Medische Biologie (PMB). Daar werken pathologen en andere stafleden waaronder (moleculair) biologen, analisten en ondersteunend personeel. De afdeling verricht naast diagnostiek ook wetenschappelijk onderzoek en geeft onderwijs. Patiënten hebben niet of nauwelijks direct contact met de afdeling PMB. Op de achtergrond speelt de afdeling echter een belangrijke rol in het ziekenhuis. De afdeling heeft twee secties: Pathologie en Medische Biologie.

De afdeling Pathologie en Medische Biologie is wat betreft de patiëntenzorg te verdelen in een afdeling cytologie, histologie en obductie. Onderstaand een duidelijke beschrijving van de verschillende bezigheden op de afdeling:

#### 1. Ontvangstruimte

De pathologie vervult een belangrijke functie op het gebied van de patiëntenzorg, het wetenschappelijk onderzoek en het onderwijs. Jaarlijks vinden er ten behoeve van de patiëntenzorg circa 30000 histologische, 20000 cytologische en 250 postmortale onderzoeken plaats. Diagnostiek vindt plaats voor het UMCG, het Scheperziekenhuis, het Wilhelminaziekenhuis alsmede enkele huisartspraktijken. Alle inzendingen krijgen hier een onderzoeksnummer.

#### 2. Uitsnijkamer

In deze ruimte wordt het weefsel volgens protocol en deels eigen inzicht van de patholoog en/of analist verder uitgesneden. Daarna wordt het verpakt in cassettes die met behulp van het, in eigen huis ontworpen, geavanceerde laboratorium management systeem worden geprint. De cassettes gaan in de VIP. Dit apparaat vervangt, in ± 8 uur, water in het weefsel door paraffine. In de uitsnijkamer worden ook de zogenaamde vriescoupes gemaakt voor de sneldiagnostiek (± 15-30 minuten).

#### 3. Paraffine

Hier wordt het weefsel ingebed in een blokje paraffine. Met behulp van een microtoom worden van het weefsel plakjes van 2 of 3 micrometer gesneden die vervolgens op objectglaasjes worden geplakt. Dit snijden vereist veel praktische vaardigheid van de analist.

#### 4. Kleuringen

Weefsels zijn meestal transparant en kleurloos. Met behulp van kleurstoffen kunnen contrasten in cellen en weefsels worden aangebracht. Bij de meeste kleurmethode

kleurstoffen chemische bindingen aan met verschillende weefselbestanddelen. Hierdoor verandert de golflengte van het licht dat door de aangekleurde weefselcomponenten passeert waardoor cellen en weefsels zichtbaar worden. Het kleuren wordt met behulp van een kleuringmachine gedaan, de Benchmark Ultra.

#### 5. Microscopie

Met behulp van een microscoop kunnen de aangekleurde preparaten worden bekeken en kan op basis van de morfologie en kleur van de cellen en weefsels een diagnose worden vastgesteld. Wanneer onduidelijkheid blijft omtrent de aandoening kan met behulp van immuunhistochemie, elektronenmicroscopie of in situ hybridisatie aanvullend onderzoek worden gedaan.

#### 6. Cytologie

Hier wordt materiaal op celniveau bekeken. Dit “turen” door een microscoop staat vooral in het teken van een “voorspelling” doen van wat men bij operatie of verder onderzoek kan aantreffen. Naast de morfologie dragen immuuncytologie, flowcytometrie en in situ hybridisatie bij aan het stellen van een diagnose.

## 2.2 Benchmark® Ultra

The Benchmark Ultra is de naam van de immunokleuring machines die op de afdeling Pathologie en Laboratoriumgeneeskunde van het UMCG worden gebruikt. Deze machines doen immunokleuringen die vroeger allemaal met de hand gedaan moesten worden. Dit scheelt veel tijd, zeker aangezien de machine ook 's nachts gedraaid kan worden zonder de aanwezigheid van medewerkers. Voor een beeld van de machines kan ik u verwijzen naar Bijlage 5.

Het Benchmark ULTRA systeem is de nieuwste generatie in de IHC- (Immuunhistochemie) en ISH-kleuringen (In Situ Hybridisatie). Doorlooptijden zijn aanzienlijk verminderd,

wat resulteert in een snellere rapportage van de resultaten aan de patiënt.<sup>1</sup>

Immuunhistochemie of IHC verwijst naar de techniek van lokaliseren van componenten (antigenen) in biologische weefsels met behulp van specifieke antilichamen. De naam is samengesteld uit "immunis", wat verwijst naar de gebruikte antilichamen, "histos" dat weefsel betekent (in vergelijking met immuuncytochemie). De techniek immuunhistochemie wordt veel gebruikt om een diagnose van abnormale cellen te stellen, zoals deze voorkomen in tumoren. Specifieke moleculaire markers zijn karakteristiek voor bepaalde cellulaire processen zoals proliferatie of de geprogrammeerde celdood. IHC wordt ook veel gebruikt in het basale onderzoek naar de distributie en lokalisatie van biomarkers en ongelijkmatig voorkomende proteïnes in verschillende delen van biologisch weefsel. Het zichtbaar maken van een antilichaam-antigeencomplex kan op verschillende manieren. Meestal wordt het antilichaam verbonden aan een enzym, zoals peroxidase, dat een kleurvormende reactie kan katalyseren.

Het BenchMark ULTRA systeem is de eerste die gelijktijdig IHC en ISH kan testen. Multi-parameter testen zoals dubbele en driedubbele kleuringen kunnen onafhankelijk van de andere dia's worden uitgevoerd, wat resulteert in snelle resultaten. The Benchmark ULTRA system kan worden gekoppeld aan een Laboratorium Informatie Systeem om informatie te beheren vanuit één computer, wat makkelijker werkt en beter is voor de veiligheid van de patiënt.<sup>2</sup>

The Benchmark ULTRA systeem heeft een nieuwe grafische gebruikersinterface (GUI), die vele voordelen biedt (echter kan deze dus niet de machine zo optimaal mogelijk indelen).

Een aantal voordelen op een rijtje:

- Elke test, wanneer dan ook: continue en random toegang tot 30 individuele kleuringkamers, de mogelijkheid tot toevoegen en verwijderen van enkele

<sup>1</sup> <http://www.ventanamed.com>

<sup>2</sup> <http://nl.wikipedia.org/wiki/Immuunhistochemie>



coupes/monsters wanneer nodig, zonder onderbreking van de machine.

- Snellere doorlooptijden: continue verwerking en loopt tot 30% sneller dan de vorige generaties.
- Monsters kunnen worden toegevoegd op elk moment voor een versnelde diagnose voor de patiënt.
- Continue informatie: intuïtieve, nieuwe software, waardoor je precies kunt zien wanneer de monsters beschikbaar zullen zijn.<sup>1</sup>



### 3 Probleemstelling en doelstelling

#### 3.1 Aanleiding Probleemstelling

De aanleiding en probleemstelling zijn uitvoerig in Hoofdstuk 1 behandeld. In deze paragraaf zal ik dit samenvattend herhalen.

Als opdracht voor mijn afstuderen heb ik geprobeerd om een waardige oplossing te vinden voor een probleem dat ontstond op de afdeling Pathologie op het UMCG. Vlak voordat ik deze opdracht heb gekregen had deze afdeling vier zo genaamde kleuringmachines aangeschaft. Hiermee hebben ze het aanbrenge van immunokleuringen op weefsels geautomatiseerd.

Het probleem van deze machines is echter dat ze ingedeeld moeten worden met de hand en aangezien de ruimte voor dispensers en coupes beperkt is, vormt zich hier de aanleiding van het probleem.

Elke machine biedt plaats aan 30 coupes en 35 dispensers. Aan standaard reagentia zijn 8 van de 35 dispenserplekken nodig. Daarnaast vergt een aantal kleuringen meer posities. Afhankelijk van deze hoeveelheid blijven er posities over om dispensers te plaatsen en verschillende kleuringen gelijktijdig te realiseren. Het is de bedoeling een algoritme te ontwikkelen voor het optimaal vullen van deze machines bij de afdeling Pathologie, rekening houdend met alle voorwaarden.

#### 3.2 Gewenste situatie

Voorop staat dat de medewerkers van de afdeling Pathologie af willen van het met de hand indelen van de kleuringmachines. Dit neemt namelijk kostbare tijd, die op een betere manier gebruikt kan worden. Hiervoor zal uiteindelijk een computerapplicatie gemaakt worden waar ik het wiskundige gedeelte voor zal ontwikkelen: het Algoritme.

Voor de Pathologen is het gemakkelijk om de verschillende kleuringen zoveel mogelijk op één machine te doen. Zo hoeven ze niet bij elke run alle dispensers te verwisselen. Daarnaast wensen de medewerkers zo weinig mogelijk zelf te doen. Ze willen op een presenteerblaadje krijgen welke kleuringen op welke machine verricht moeten worden, het liefst zonder dat ze er ook maar iets voor hoeven doen. Natuurlijk is dit laatste niet realiseerbaar, maar het is wel mogelijk om ze zo weinig mogelijk verrichtingen handmatig te laten doen.

#### 3.3 Voorwaarden

Een aantal voorwaarden heb ik al genoemd. Zo heeft elke machine plek voor maximaal 30 coupes en 35 dispensers. Van de 35 dispensers zijn er standaard al 8 bezet door standaardreagentia, wat betekent dat er 27 plekken overblijven.

Om duidelijkheid te verschaffen in mijn rapport heb ik de term kleursoort ingevoerd. Er zijn vijf verschillende kleursoorten: Ultraview, I-view, Eber, Rood en Dual-Sish. Deze kleursoorten vergen elk een verschillend aantal extra dispensers. Hierbij moet vermeld worden dat Eber en Dual-Sish eigenlijk geen kleursoorten zijn, maar voor de duidelijkheid in dit rapport wel zo genoemd worden. Deze twee zijn namelijk gewoon de naam van de kleur (zie Bijlage 1, lijst van immunokleuringen).

Elk soort kleuring vergt zoals gezegd een verschillend aantal dispensers. Zo neemt een rode kleuring eenmalig 5 extra dispenserplekken in. Dit betekent niet dat twee rode kleuringen 10 extra dispenserplekken innemen! Dit blijven er dan 5. Ultraview kleuringen nemen geen extra dispenserplekken (deze zijn namelijk al verrekend in de 8 standaardreagentia die in iedere machine zitten). I-view kleuringen nemen 8 extra posities, Eber 9, Rood 5 en Dual-

Sish 17. Deze gegevens kunt u in een overzichtelijke tabellen terugvinden in bijlage 2 en 3. Van bijna iedere kleur is slechts één dispenser aanwezig. Dit maakt het onmogelijk om een kleur te verdelen over meerdere machines. De kleuring met naam "lg Kappa.br/Lamda.rd" neemt twee dispenserposities in en moet ook rood gekleurd worden. Dit kan in de toekomstige computerapplicatie simpel opgelost worden door een Kappa Rood en een Lambda Rood te creëren waar vervolgens elk de helft van het aantal coupes achter vermeld moet worden. Er lijkt dan geen zekerheid te zijn dat de kleuren ook daadwerkelijk in dezelfde machine ingedeeld worden, maar gelukkig is deze kleuring Rood en zullen ze dus ten aller tijde in Machine 3 geplaatst worden (zie algoritme). Er zijn ook twee kleuren, Melan A en S-100, die soms wel en soms niet rood gekleurd moeten worden. Ook dit is simpel op te lossen in het programma door ook een "Melan A Rood" en een "S-100 Rood" te creëren.

programma kan schrijven dan dat ik dat kan. Daarnaast kan ik waarschijnlijk niet voldoen aan alle eisen die de werknemers van de afdeling Pathologie aan de applicatie zouden stellen. Het is ook van belang dat er in een dergelijke applicatie veel tijd en energie gestoken wordt, om er voor te zorgen dat het in één keer zo goed mogelijk neergezet wordt. Tijdens mijn afstuderen beschik ik niet over deze benodigde tijd.

Het algoritme wordt in een pseudotaal geleverd, zodat hij door iedere programmeur begrepen kan worden. Ik heb er voor gekozen het niet in een computertaal van een bepaald programma te schrijven, omdat het nog niet bekend is in welk programma de uiteindelijke applicatie het beste tot zijn recht zal komen. Daarnaast is het algoritme ook in verhaallijn in het rapport opgenomen worden om zo de nodige ondersteuning te geven.

### 3.3 Vraagstelling

Uitgaande van een nachtrun (rond 16.30 uur wordt de machine gestart, waarna deze zelfstandig de ingestelde kleuringen aanbrengt) en een 2<sup>e</sup> run overdag (op werkdagen wordt de machine overdag ook gebruikt): Hoe vullen we zo efficiënt mogelijk de 4 machines? Welke kleuring verrichten we op welke machine? Is het mogelijk hier een algoritme voor te schrijven die in de toekomst in een computerapplicatie verwerkt kan worden?

### 3.4 Doelstelling

De doelstelling is om een algoritme te ontwikkelen die in de toekomst in een goed werkende en doordachte computerapplicatie verwerkt kan worden. Ik kies ervoor om deze applicatie niet zelf te ontwikkelen, aangezien ik geen volleerd programmeur ben. Ik ben in de veronderstelling dat een goede programmeur een veel beter en efficiënter

## 4 Opzet en uitvoering

### 4.1 Aanleiding en doel van het onderzoek

De aanleiding van het onderzoek was mij tijdens het eerste gesprek al duidelijk. De afdeling Pathologie had vier nieuwe machines aangeschaft, zodat de kleuringen sneller gedaan konden worden. Ook kunnen de machines 's nachts gewoon gedraaid worden, omdat er niets meer ingesteld hoeft te worden zodra de machines ingesteld zijn. Het probleem van deze machines is dat ze een beperkte inhoud hebben. Er kunnen 30 coupes en 35 dispensers in. Van bijna iedere kleur is slechts één dispenser (met antilichamen) aanwezig. Voor de Pathologen is het lastig om de machines zo optimaal mogelijk in te delen. Dit komt neer op 120 coupes verdelen over 4 machines zonder dat het aantal dispensers groter wordt dan 35 per machine. Momenteel hebben ze aardig door hoe de machines het beste ingedeeld kunnen worden, maar ze hebben liever een computerapplicatie die dit voor hun bepaald.

### 4.2 Inhoudelijke oriëntatie

Ik ben te vroeg begonnen met mijn inhoudelijke oriëntatie. Het idee uit mijn eerste onderzoeksvoorstel om een programma te schrijven in VBScript of VBA (Visual Basic for Applications) bleef in mijn hoofd rondspoken. Ik heb aan het begin van mijn afstudeeropdracht best veel tijd gestoken in het begrijpen van deze programmeertaal voor Officepakketten. Boeken die ik (deels) doorgenomen heb zijn vermeld in de bronvermelding. Enige tijd later kwam ik er achter dat het begrijpen van het totale indelingsprobleem nogal wat tijd zou gaan vergen en dat het niet zo simpel was als dat het leek. Doordat het probleemcomplex was, ben ik niet meer toegekomen aan het programmeren in VBScript. Aangezien er geen digitale of schriftelijke bronnen zijn waar een zelfde indelingsprobleem in besproken werd, was

inhoudelijke oriëntatie hiervan in de vorm van gericht vragen stellen aan de werknemers van de afdeling Pathologie. Over de werking van de machines kon ik wel bronnen op internet vinden. Op de website van de uitvinders van de machines wordt enige informatie over deze machines prijsgegeven. Zie digitale bron nummer 1. Voor het knapzakprobleem, welke onderstaand beschreven is, heb ik veel literatuur geleend van de Universitaire Bibliotheek in Groningen. Jammer genoeg bleek ook het knapzakprobleem niet in verband te brengen met mijn indelingsprobleem. De literatuur die ik gelezen heb kunt u terugvinden in de bronvermelding.

### 4.3 Aanpak van het onderzoek

Veel vragen stellen, ideeën opdoen en uitproberen. Dat typeert vooral het begin van het oplossen van het probleem.

Veel vragen stellen was gedurende de hele periode zeer nodig. Allereerst al omdat ik geen verstand heb van Pathologie en de machines. Daarnaast omdat ik niet alle informatie verkreeg die ik nodig had. Hiermee wil ik zeggen dat ik de informatie die voor een Patholoog heel logisch is uit hem/haar heb moeten weten te krijgen. Zij zullen dit namelijk niet cadeau geven aangezien het voor hen vanzelfsprekend lijkt. Zo kwam ik er bijvoorbeeld achter, nadat ik zelf verzonnen had dat de verschillende soorten kleuringen het beste op verschillende machines gedaan konden worden, dat de Pathologen dit ook doen. Ideeën opdoen was soms niet makkelijk. Omdat het probleem complex is en op geen manier op bestaande problemen lijkt, is het lastig een vergelijkbaar probleem en oplossing daarvan te zoeken wat door niet al te veel aanpassingen als (deel)oplossing voor het machine indelingsprobleem kan dienen.

Als er dan een (deels) vergelijkbaar probleem en oplossing gevonden is, zal het ook uitgeprobeerd/getest moeten

worden. Bij het uitproberen kwam ik er iedere keer achter dat mijn probleem niet op een dergelijke manier op te lossen was. Dit werkt best frustrerend, maar zorgt er ook voor dat ik nog gedreven werd om een oplossing te vinden.

Onderstaand zal ik de bruikbare en niet bruikbare handelingen die ik verricht heb toelichten:

#### *Niet bruikbare handelingen*

Verscheidene methoden waarvan ik hoopte dat deze tot een gewenste oplossing zouden leiden bleken niet bruikbaar. Beginnend met een vrij simpele methode: Lineair Programmeren. Op lineair programmeren was ik vrij gauw uitgekeken, omdat alle voorwaarden van het probleem simpelweg niet verwerkt kunnen worden in een stelsel van lineaire vergelijkingen. Een Roosterprobleem leek mij op het eerste gezicht een uitstekende manier om het probleem op te lossen. Je kunt gemakkelijk over vier machines het overzicht bewaren. Je kunt een maximum stellen van 35 voor het aantal dispensers. Je kunt de soorten kleuringen er eventueel ook in verwerken. Het enige probleem is dat je geen limiet kunt stellen aan het maximaal aantal glaasjes dat geplaatst mag worden. In Bijlage 4 heb ik dit duidelijk proberen te maken met behulp van een schema. Door middel van het maken van een graaf is het niet mogelijk de indeling van de machines te optimaliseren. Allereerst vormt het maken van de graaf al een probleem en vervolgens liep ik tegen het probleem dat er alleen algoritmen zijn voor koppelingsproblemen, bijvoorbeeld voor het zoeken van een lichtste complete koppeling. Dit is niet van toepassing op het probleem wat ik op moet lossen. Door een medestudent ben ik in een later stadium geweest op het Bin Packing Problem oftewel:

#### *Het Knapzakprobleem*

Gegeven een verzameling van objecten, elk met gewicht en waarde, bepaal welke (deel)verzameling van objecten meegenomen wordt in de knapzak, zodat het totale gewicht onder de limiet blijft en de totale waarde gemaximaliseerd wordt.<sup>3</sup> Er bestaan geen betere methoden

om dit probleem op te lossen dan Brute Force (alle  $2^x$  mogelijkheden proberen). Ook is er geen algoritme bekend die altijd het juiste antwoord geeft en minder dan exponentiële tijd nodig heeft. Er zijn wel betere benaderingen van het knapzakprobleem bekend. Hier zijn verschillende heuristieken voor bedacht die niet zomaar zijn verzonden. Hier zit vaak een hele gedachte achter en per probleem is de heuristiek weer heel verschillend. Ik dacht het machine indelingsprobleem te kunnen koppelen met het Knapzakprobleem door de coupes gelijk te stellen aan de waarden en de dispensers zijn dan het gewicht.. De totale waarde (totaal aantal coupes) moet dan gemaximaliseerd worden met de extra voorwaarde dat het niet groter mag zijn dan 30 en het totale gewicht (totaal aantal dispensers) moet onder de limiet 35 blijven. Het bleek echter niet makkelijk te zijn om mijn probleem om te zetten in een knapzakprobleem. Er moesten te veel aanpassingen gedaan worden die het Knapzakprobleem uitvoeriger en dus niet gemakkelijker maakten. Ook moest ik vier knapzakken creëren (4 machines) die allemaal iets van elkaar verschillen. Aangezien het knapzakprobleem van zichzelf al een moeilijk probleem is, waar geen algoritme van bekend is die altijd het juiste antwoord geeft, leek het mij verstandig hier niet verder op door te gaan. Ik heb veel boeken en internetpagina's opgezocht en gelezen, maar blijf erbij dat ik er verstandig aan gedaan heb om het Knapzakprobleem naast mij neer te leggen. Ondanks dat het Knapzakprobleem erg interessant is en het erg veel op het machine indelingsprobleem lijkt.

#### *Bruikbare handelingen*

Uiteindelijk ben ik simpelweg begonnen met het schrijven van een zelfgemaakt algoritme voor het optimaal vullen van de kleuringmachines. Door alle voorwaarden netjes op papier te zetten en na te gaan welke handelingen de Pathologen verrichten om de indeling van de machines te maken kon ik een begin van mijn algoritme op papier krijgen.

Het volgende hoofdstuk geeft een toelichting van de handmatige ontwikkeling van het algoritme.

<sup>3</sup> <http://nl.wikipedia.org/wiki/Knapzakprobleem>

## 5 Resultaten

### 5.1 Ontwikkeling van het algoritme

Het was voor mij eerst van belang alles goed op een rijtje te hebben. Alle voorwaarden (§3.3) heb ik duidelijk op papier gezet en laten controleren door een Patholoog.

Vervolgens ben ik na gegaan hoe ik dit in een algoritme krijgen kon. Ieder idee dat bij mij opkwam heb ik in verhaallijn op papier gezet. Ik heb niet ieder idee in algoritmetaal gezet. Dit zou namelijk gezorgd hebben voor onnodige tijdsverspilling.

Omdat op iedere machine dagelijks ultraview kleuringen gedaan worden, laten de Pathologen de 8 dispensers die hiervoor nodig zijn standaard in de machine. Vertaald naar het algoritme betekend dit dat er niet 35 dispenserplekken beschikbaar zijn, maar 27. Dit rekent gemakkelijker en hierdoor hoef je nooit rekening te houden met de standaardreagentia voor de Ultraview kleuring.

Vervolgens bedacht ik dat het voor de indeling van de machines het beste is dat elk soort kleuring op een verschillende machine gedaan wordt. Hierdoor voorkom je dat je veel extra dispenserplekken moet gebruiken voor slechts een paar kleuringen. Stel dat er 4 rode kleuringen en 5 Dual Sish kleuringen in één machine worden geplaatst, dan zijn er slechts 9 coupes geplaatst. Daar tegenover staat dat er al 22 extra dispenserplekken gebruikt zijn. Er zijn dus nog 21 coupes plekken over en 6 (35 - 22 - 8 ultra view) dispenserplekken. Dat is natuurlijk verre van optimaal! Om dit te voorkomen is het dus verstandig om op iedere machine één van de vier soorten kleuringen te verrichten. Ultraview niet meegeteld omdat deze al standaard op elk machine geplaatst is.

Door slim na te denken wou ik de rest van het algoritme ontwikkelen, zodat die zou leiden tot een optimale indeling van de machines. Het begin was er, maar het vervolg moest nog komen. Dit vervolg (de kern van het algoritme) heeft voor veel kopzorgen gezorgd. Onderstaand zal ik duidelijk maken welke procedure ik gevolgd voor de ontwikkeling

van mijn definitieve algoritme. Deze procedure heeft ertoe geleid dat ik tot mijn definitieve algoritme ben gekomen, maar veel stappen zul je in het definitieve algoritme niet meer terug zien.

#### *Algoritme 1*

Mijn gedachte bij mijn eerste algoritme was als volgt: Als er een kleur is waar meer dan 10 coupes gekleurd moeten worden, dan is dit niet gunstig voor de indeling van de machine. Er worden dan veel coupes, maar slechts enkele dispenserplekken bezet. Hierdoor zit de machine snel op de limiet van het aantal coupes terwijl er nog veel dispenserplekken vrij zijn! Om dit te voorkomen deel ik alle kleuren waarvan het aantal coupes groter is dan 10 door 2. Is het aantal coupes oneven? Dan wordt het gedeeld door 2, waarbij het ene deel afgerond wordt naar boven en het andere naar beneden. Pas als van alle kleuren het aantal coupes kleiner is dan 10 ga ik ze indelen in de machines. Daarbij neem ik de kleur met het grootste aantal coupes en plaats deze in de machine met het kleinste aantal dispensers. Hierbij zorg ik ervoor dat er zoveel mogelijk coupes in de machine komen waar het limiet van dispenserplekken het eerste bereikt wordt. Op het moment dat een opgedeelde kleur toch weer samen in een van de machines geplaatst wordt dan ontstaat er een nieuw probleem. Eenzelfde dispenser voor deze kleur wordt dan namelijk twee keer in de machine geplaatst, terwijl er slechts één nodig is!

#### *Algoritme 2*

Er was nog een reden dat ik niet tevreden was over voorgaande gedachte, want het gaat er uiteindelijk om dat er 30 coupes in elke machine komen en het maakt niet uit hoeveel dispenserplekken er daarbij overblijven. Het idee om kleuren met een coupesgrootte van 10 of meer door tweeën te delen heb ik dus geschrappt. Maar als een kleur in geen van de machines geplaatst kan worden omdat daar geen ruimte meer voor is, leek het mij wel handig om de

betreffende kleur door tweeën te delen en weer te rangschikken in de lijst van de kleuren die nog niet ingedeeld zijn. Zo zorg ik ervoor dat deze kleur alsnog ingedeeld kan worden, echter dan in twee delen! Zo had ik een mooi algoritme op papier, maar kreeg ik van de afdeling Pathologie te horen dat er van (bijna) iedere kleur slechts één dispenser aanwezig is. Alleen van de kleuren die heel veel voorkomen hebben ze de beschikking over twee dispensers.

### *Algoritme 3*

Door dit negatieve bericht kon ik de kern van mijn algoritme helemaal vernieuwen, maar het idee om de kleur met de meeste coupes te plaatsen in de machine met de minste dispensers bleef bestaan. Toch heb ik dit ook aangepast, omdat een machine bijvoorbeeld 4 dispenserplekken en 8 coupesplekken de kleur met de meeste coupes minder nodig heeft dan een machine met 5 dispenserplekken en 15 coupesplekken. Dit aangezien het vol krijgen van de tweede machine moeilijker is dan de eerste. Dit heb ik opgelost door de kleur met de meest aantal coupes in de machines met de grootste coupes/dispenser verhouding te plaatsen. Dit betekend dat in het voorbeeld de kleur met de meeste coupes in de machine met 5 dispenserplekken gezet wordt, omdat  $15/5$  groter is dan  $8/4$ . Met deze gedachte op papier dacht ik weer een aardig algoritme op de been te hebben, maar tevreden was ik nog niet!

### *Algoritme 4*

Het laatste idee voordat ik naar mijn definitieve algoritme toe kon werken was het volgende:

Als in een machine, na de plaatsing van de 4 soorten kleuringen, het aantal dispenserplekken groter is dan het aantal coupesplekken dan maakt het niet uit wat hier nog bij ingedeeld wordt! Dit betekend dat als ik er voor zorg dat het aantal dispenserplekken in iedere machine groter is dan het aantal coupesplekken dat het restant van de coupes willekeurig in de machine geplaatst kan worden. Alle kleuren die nog ingedeeld moet worden in de machines, worden gesorteerd van klein (minste coupes)

naar groot (meeste coupes) en de kleuren met één coupe worden tijdelijk buiten beschouwing gelaten.

Als op dit moment in een van de machines het aantal dispenserplekken kleiner is dan het aantal coupesplekken, dan wordt de kleur met het meeste aantal coupes in de betreffende machine geplaatst, onder de voorwaarde dat dit past. Als dit niet past wordt de kleur met het op 1 na meeste aantal coupes geprobeerd, etc.

Als deze handeling ervoor gezorgd heeft dat in alle machines het aantal dispenserplekken groter is dan het aantal coupesplekken, dan kan het restant van de plekken in de machines ingedeeld worden met behulp van backtracking: [Het voordeel van backtracking is dat alle deelverzamelingen kunnen worden bekeken, maar ze niet allemaal hoeven te worden gegenereerd. Een deelverzameling van  $\{s_1, s_2, \dots, s_n\}$  met som  $s'$  hoeft niet verder uitgebreid te worden als  $s'+s_{(i+1)} > A, B, C$  of  $D$  (omdat er dan zeker is dat de rest van de kleuren ook niet past). Of als  $s'+\text{SOM}(j=i+1 \text{ tot } n)$  van  $s_j < A, B, C$ , of  $D$  (omdat dan zeker is dat de gehele verzameling in de machine past). Dit aangezien verzameling  $S$  olopemd gesorteerd is.]

Overgebleven is een verzameling van het aantal coupes  $S = \{s_1, s_2, \dots, s_n\}$  van positieve gehele getallen groter dan 1 en vier gehele getallen  $A, B, C$  en  $D$  (het aantal overgebleven coupesplekken van de machines). Met behulp van backtracking proberen we nu vier deelverzamelingen van  $S$  te vinden waarvan de som gelijk is aan  $A, B, C$  en  $D$ , want dan zijn alle machines gevuld met 30 coupes.

We bekijken alle deelverzameling van  $A, B, C$ , en  $D$ . Vervolgens wordt er bekeken met welke combinatie we de meeste volledige deelverzamelingen (precies gelijk aan  $A, B, C$  en/of  $D$ ) kunnen genereren en dit wordt als oplossing gebruikt. Hierbij kan het zijn dat er machines niet ingedeeld kunnen worden met een volledige deelverzameling. Voor deze machines wordt bekeken of zij wel volledig kunnen worden ingedeeld met de overgebleven verzameling als we respectievelijk 1, 2, 3, ...,  $N$  coupesplekken weglaten. Deze machines worden op het eind zoveel mogelijk aangevuld met de buiten beschouwing gelaten kleuren met één coupe. Zo heb ik ervoor gezorgd dat in iedere machine zo weinig mogelijk coupesplekken ongebruikt blijven.



## 5.2 Het definitieve algoritme

Met de gedachte dat mijn vierde algoritme veel te ingewikkeld is en het veel gemakkelijker moest kunnen heb ik een laatste poging gewaagd om een beter algoritme te schrijven. Onderstaand zal ik de probleembeschrijving herhalen en vervolgens zal ik het algoritme van begin tot eind met woorden omschrijven/toelichten. In paragraaf 5.3 staat het algoritme in de taal zoals hij geleverd wordt aan een programmeur en in paragraaf 5.4 is het algoritme getest.

### *Probleembeschrijving*

De afdeling Pathologie heeft 4 machines aangeschaft die immunokleuringen verrichten, wat voorheen door Pathologen met de hand gedaan werd. In elk van deze machines kunnen 30 coupes (glaasjes met weefsel) en 35 dispensers (buisje met benodigde stof om een kleuring te verrichten) geplaatst worden. Elke kleur heeft minstens één dispenser nodig met een antilichaam, ongeacht het aantal coupes. De kleuren kunnen onderverdeeld worden in 5 soorten die elk een bepaald aantal extra dispenserplekken in beslag neemt:

Ultraview: De meeste kleuren zijn Ultraview. Als er een Ultraview kleuring op een machine gedaan wordt, neemt dat eenmalig 8 extra dispenserplekken in beslag.

I-view: Deze kleuringen nemen eenmalig 8 extra dispenserplekken in beslag.

Eber: Dit is ook de naam van de kleur. Er zijn verder geen kleuren die onder Eber vallen. Deze kleuring neemt eenmalig 9 extra dispenserplekken in beslag.

Rood: Deze kleuringen nemen eenmalig 5 extra dispenserplekken in beslag.

Dual-Sish: Ook dit is de naam van de kleur zelf. Er zijn verder geen kleuren die onder deze naam vallen. Deze kleuring neemt eenmalig 17 extra dispenserplekken in beslag! Hier vormt zich het grootste probleem qua indeling van de machines.

We hebben vier machines met elk hun vaste soort kleuringen:

Machine 1: Ultraview en I-view

Machine 2: Ultraview en Eber

Machine 3: Ultraview en Rood

Machine 4: Ultraview en Dual-Sish

Aangezien iedere machine ultraview kleuringen verricht is het gemakkelijker als deze dispensers standaard in iedere machine geplaatst worden. Het komt er dus op neer dat we gaan rekenen met 27 vrije posities voor dispensers in iedere machine. Nu is het de kunst om de machines zo optimaal mogelijk in te delen, wat betekend dat in iedere machine zoveel mogelijk coupes geplaatst moeten worden.

### *Algoritme uitleg*

#### **Begin**

Geef aan welke kleuringen er gedaan moeten worden, door in te vullen hoeveel glaasjes daarvoor geplaatst moeten worden.

Elke machine heeft 27 overgebleven en dus beschikbare plekken voor dispensers. Er zijn namelijk al 8 bezet door Ultraview dispensers. Er zijn in iedere machine 30 beschikbare plekken voor coupes.

Onder de volgende voorwaarden wordt het aantal beschikbare plekken voor dispensers in de betreffende machine aangepast:

Zijn er...

- I-view kleuringen? Zo ja, dan heeft Machine 1 nog 19 plekken voor dispensers;
- Eber kleuringen? Zo ja, dan heeft Machine 2 nog 18 plekken voor dispensers;
- Rode kleuringen? Zo ja, dan heeft Machine 3 nog 22 plekken voor dispensers;
- Dual-Sish kleuringen? Zo ja, dan heeft Machine 4 nog 10 plekken voor dispensers.

Als dit aangepast is, zullen de bovenstaande 4 soorten kleuringen ook daadwerkelijk in de betreffende machine worden ingedeeld. Voor iedere verschillende kleuring (met één of meerdere glaasjes) wordt nog één dispenserplek in de betreffende machine bezet. Het aantal beschikbare

coupeplekken neemt af tot “30 – aantal coupes van de kleur” onder de voorwaarde dat:

- Dispenser (X) = aantal overgebleven dispenser-plekken in Machine X  $\geq 1$  en
- Coupes (X) = aantal overgebleven coupeplekken in Machine X  $\geq$  “aantal kleur”.

Waarbij “aantal kleur” het aantal coupes van de kleur betreft en X = 1, 2, 3 of 4.

Vervolgens moeten alle kleuren die nog niet ingedeeld zijn in een machine worden gesorteerd van klein (minste coupes) naar groot (meeste coupes).

### Herhaal

Het programma moet nu de 4 machines zo optimaal mogelijk in gaan delen.

Als op dit moment in een van de machines het aantal dispenserplekken kleiner is dan het aantal coupeplekken, dan wordt de kleur met het meeste aantal coupes in deze machine geplaatst, onder de voorwaarde dat dit past. Als dit niet past, wordt de kleur met het op één na meeste aantal coupes geprobeerd, etc.

Als deze bewerking ervoor gezorgd heeft dat in alle machines het aantal dispenserplekken groter is dan het aantal coupeplekken, dan kan het restant van de plekken in de machines als volgt ingedeeld worden:

Neem de machine met de meeste aantal coupeplekken en plaats hier de overgebleven kleur met de meeste aantal coupes. Herhaal deze bewerking totdat geen van de overgebleven kleuren meer geplaatst hoeft/kan worden. De kans dat een machine niet vol komt te zitten is slechts aanwezig als:

- Er geen 120 coupes beschikbaar zijn;
- Er geen of bijna geen “klein aantal” (1 of 2) coupes van één kleur gekleurd hoeft te worden.

Aangezien deze gebeurtenis (bijna) nooit voorkomt kunnen we deze kans verwaarlozen.

Er bestaat een kans dat een machine uitvalt en dus niet gebruikt kan worden. Dit zou betekenen dat niet alle soorten kleuringen kunnen worden ingedeeld, want als Machine 3 uitvalt worden de rode kleuringen niet ingedeeld! Er moet een mogelijkheid komen in de

computerapplicatie om machines uit te schakelen, zodat ze niet meedoen in de berekening. Daarnaast moeten het aantal coupeplekken en dispenserplekken ook handmatig aangepast kunnen worden. Zo kunnen Pathologen door slim denken (wat een applicatie niet kan) alsnog kleuringen op een andere machine verrichten.

Het zou ook voor kunnen komen dat een machine geheel wordt ingedeeld door slechts één of twee kleuren! Dit kan de Patholoog dan handmatig doen en de machine in de applicatie uitschakelen.

### Eind

Als resultaat verschijnt de indeling van alle vier machines, waarbij aangegeven wordt hoeveel glaasjes van welke kleuring in Machine 1, 2, 3 en 4 geplaatst moeten worden. Daarnaast wordt aangegeven hoeveel vrije plaatsen (dispensers en coupes) er zijn in iedere machine.

### 5.3 Algoritme “Immunocalculator”

Het definitieve algoritme heb ik de naam “Immunocalculator” gegeven. Ik geef nu eerst een uitleg van een aantal begrippen uit het algoritme.

Iedere kleur krijgt een eigen nummer  $i = 1, 2, \dots, N$

Kleur(i) = Naam van de kleur met nummer i

Aantal(i) = Aantal coupes van kleur i

Soort(i) = Soort kleur van kleur i = Ultraview / I-view / Eber / Rood / Dual-Sish

D = Dispenser

aantalD(M) = totaal aantal overgebleven dispenserplekken in Machine M, met M = 1, 2, 3 of 4

Startwaarde aantalD(M) = 27

C = Coupe

aantalC(M) = totaal aantal overgebleven coupeplekken in Machine M, met M = 1, 2, 3 of 4

Startwaarde aantalC(M) = 30

i	Kleur	Soort kleur	Aantal Coupes (1, 2, ..., 30)
1	Melan A	Ultra-view	4
2	ZAP-70	Ultra-view	7
3	1C2	I-view	3
4	Amyloid Bèta	I-view	1
5	Eber	Eber	3
6	CD 4	Rood	5
7	Melan A	Rood	1
8	Dual Sish	Dual Sish	6

**Tabel 1** voorbeeldtabel van de kleuren

Machine X met  $X = 1, 2, 3$  of  $4$  is het nummer van de Machines op de afdeling Pathologie van het UMCG van links naar rechts.

Tabel 1 geeft een voorbeeld van een aantal kleuren, het soort kleur en een  $x$  aantal coupes dat gekleurd moet worden. Wellicht wordt het geheel hierdoor iets duidelijker.

Vanaf de volgende pagina begint het algoritme, geschreven in pseudotaal (formuleringen in het Nederlands die ik gebruik om het algoritme te schrijven) en in slechts één kolom, omdat het algoritme dan beter tot zijn recht komt.

## Begin algoritme

### Machine 1

Voor  $i = 1$  tot  $n$

Als Soort( $i$ ) = I-view

Als aantal( $i$ )  $\leq$  aantalC( $1$ ) en aantalD( $1$ )  $> 8$

aantalD( $1$ ) := aantalD( $1$ ) - 1

aantalC( $1$ ) := aantalC( $1$ ) - aantal( $i$ )

kleur( $i$ ) in Machine 1

Anders kleur( $i$ ) buiten beschouwing laten

Als {na voorgaande bewerking} aantalD( $1$ )  $< 27$

aantalD( $1$ ) := aantalD( $1$ ) - 8

### Machine 2

Als aantal(Eber)  $> 0$

aantalD( $2$ ) := 18

aantalC( $2$ ) := aantalC( $2$ ) - aantal(Eber)

Eber in Machine 2

### Machine 3

Voor  $i = 1$  tot  $n$

Als Soort( $i$ ) = Rood

Als aantal( $i$ )  $\leq$  aantalC( $3$ ) en aantalD( $3$ )  $> 5$

aantalD( $3$ ) := aantalD( $3$ ) - 1

aantalC( $3$ ) := aantalC( $3$ ) - aantal( $i$ )

kleur( $i$ ) in Machine 3

Anders kleur( $i$ ) buiten beschouwing laten

Als {na voorgaande bewerking} aantalD( $3$ )  $< 27$

aantalD( $3$ ) := aantalD( $3$ ) - 5

#### Machine 4

---

Als aantal(Dual Sish) > 0  
aantalD(4):=10  
aantalC(4):=aantalC(4)-aantal(Dual Sish)  
Dual Sish in Machine 4

---

*{We hebben nu de speciale soorten kleuren ingedeeld in Machines 1 t/m 4. Het aantal dispenserplekken en coupesplekken zijn hierbij al aangepast. Eber en Dual Sish hebben een andere bewerking, aangezien dit zelf kleuren zijn.}*

Sorteer nu alle kleuren die nog niet ingedeeld zijn van klein (minste coupes) naar groot (meeste coupes). Dit kan met allerlei sorteeralgoritmen die voor het oprapen liggen. De keuze hiervan laat ik over aan de programmeur.

We houden over de kleuren die gesorteerd zijn met nieuwe nummer  $j = 1, 2, \dots, m$

#### Kern algoritme (herhaal statement)

---

Voor  $i = 1$  tot 4 herhaal  
  Als aantalD(i) < aantalC(i)  
    Voor  $j = m$  tot 1  
      Als aantal(j) <= aantalC(i)  
        aantalC(i) := aantalC(i) - aantal(j)  
        *{Overgebleven kleur met meeste coupes komt in Machine i}*  
        aantalD(i) := aantalD(i) - 1  
        kleur(j) in Machine i  
      Anders kleur(j) terug plaatsen in verzameling kleuren en bovenstaande bewerking opnieuw uitvoeren met kleur(j-1).  
    Stop als aantalC(i) <= aantalD(i)

*{We hebben ervoor gezorgd dat in alle machines het aantal overgebleven coupesplekken kleiner of gelijk is aan het aantal overgebleven dispenserplekken.}*

---

Voor  $j = m$  tot 1 herhaal  
  Voor  $i = 1$  tot 4  
    Als aantalC(i) is maximaal en aantalC(i) >= aantal(j)  
      aantalC(i) := aantalC(i) - aantal(j)  
      *{kleur met grootste aantal coupes komt in de machine met het meeste aantal coupesplekken}*  
      aantalD(i) := aantalD(i) - 1  
      kleur(j) in Machine i  
    Anders kleur(j) buiten beschouwing laten  
  Stoppen als er geen kleur meer ingedeeld hoeft te worden of als bij alle machines het volgende geldt:  
  aantalC(i) = 0 of aantalD(i) = 0

*{Alle machines zitten nu zo vol mogelijk. Voor elke kleur is bekeken of die nog in een van de machines past. Past dit niet dan wordt deze kleur buiten beschouwing gelaten.}*

---

#### Eind algoritme

---

Als resultaat verschijnt de indeling van alle vier machines, waarbij aangegeven wordt hoeveel glaasjes van welke kleuring in Machine 1, 2, 3 en 4 geplaatst moeten worden. Daarnaast wordt aangegeven hoeveel vrije plaatsen (dispensers en coupes) er zijn in iedere machine.

---

## 5.4 Het algoritme testen

Om enigszins iets over de bruikbaarheid van het algoritme te kunnen zeggen, moet het natuurlijk getest worden. Ik heb het algoritme getest aan de hand van werkelijke waarden van de afdeling Pathologie. Ik heb de Pathologen gevraagd om een aantal dagen bij te houden welke en hoeveel kleuringen er iedere dag gedaan werden op iedere machine. Helaas was er een aantal van deze dagen een van de machines defect. Hierdoor werd de betreffende data onbruikbaar. Daarnaast zijn de machines niet bij iedere run helemaal vol. Natuurlijk kan mijn algoritme hiermee omgaan, maar dan komt hij niet helemaal tot zijn recht.

Gelukkig was er wel een dag dat alle vier machines vol waren. De data van deze dag heb ik gebruikt om het algoritme te testen, want juist wanneer in alle vier machines 30 coupes geplaatst kunnen worden is de kans het grootst dat het algoritme faalt. Onderstaand toon ik aan dat de test goed verlopen is en dat het algoritme dus juist werkt. Ik heb de kleuren in dit geval geen nummers gegeven, om duidelijk onderscheid te krijgen tussen het aantal coupes en de kleurnamen. Ook vond ik het overbodige informatie om alle kleuren hun werkelijke naam te geven.

In tabel 3 staan de bijbehorende indeling van de machines weergegeven.

Kleurnaam	aantal	Soort	Kleurnaam	aantal	Soort	Kleurnaam	aantal	Soort
a	4	Ultraview	ar	1	Ultraview	i	2	Ultraview
aa	2	Ultraview	at	2	Ultraview	j	2	Ultraview
ab	1	Ultraview	au	5	Ultraview	k	1	Ultraview
ac	1	Ultraview	av	2	Ultraview	l	2	Ultraview
ad	1	Ultraview	aw	2	Ultraview	m	3	Ultraview
ae	1	Rood	ax	2	Ultraview	n	1	Ultraview
af	5	Rood	ay	2	Ultraview	o	3	Ultraview
ag	4	Rood	az	3	Ultraview	p	1	Ultraview
ah1	2	Rood	b	4	Ultraview	q	4	Ultraview
ah2	2	Rood	ba	4	Ultraview	r	1	Ultraview
ai	2	Ultraview	bb	2	Ultraview	s	1	Ultraview
aj	2	Ultraview	bc	3	Ultraview	t	1	Ultraview
ak	2	Ultraview	c	5	Ultraview	u	2	Ultraview
al	2	Ultraview	d	3	Ultraview	v	1	Ultraview
am	2	Ultraview	e	4	Ultraview	w	1	Ultraview
an	1	Ultraview	Eber	4	Eber	x	1	Ultraview
ao	1	Ultraview	f	2	Ultraview	y	1	Ultraview
ap	1	Ultraview	g	2	Ultraview	z	1	Ultraview
aq	1	Ultraview	h	2	Ultraview			

Tabel 2 in te delen kleuren

Machine 1	aantal Coupes	Aantal Disp.	Machine 2	aantal Coupes	aantal Disp.	Machine 3	aantal Coupes	Aantal Disp.	Machine 4	aantal Coupes	aantal Disp.
I-view		0	Eber	4	9	Rood		5	Dual-Sish	0	0
au	5	1	c	5	1	ae	1	1	b	4	1
e	4	1	a	4	1	ah1	2	1	ba	4	1
az	3	1	m	3	1	ah2	2	1	q	4	1
bc	3	1	aj	2	1	ag	4	1	d	3	1
aa	2	1	at	2	1	af	5	1	ai	2	1
ak	2	1	ay	2	1	o	3	1	am	2	1
av	2	1	h	2	1	al	2	1	ax	2	1
bb	2	1	u	2	1	aw	2	1	g	2	1
i	2	1	ao	1	1	f	2	1	l	2	1
ab	1	1	k	1	1	j	2	1	ad	1	1
an	1	1	s	1	1	ac	1	1	aq	1	1
ar	1	1	x	1	1	ap	1	1	p	1	1
r	1	1				n	1	1	v	1	1
w	1	1				t	1	1	z	1	1
						y	1	1			
<b>Over:</b>	0	13	<b>Over:</b>	0	6	<b>Over:</b>	0	7	<b>Over:</b>	0	13

**Tabel 3** indeling van de machines

In bovenstaande tabel ben ik te werk gegaan volgens het algoritme. I-view kleuren waren er niet, dus heb ik eerst de kleur Eber ingedeeld in Machine 2. Hier waren 4 coupes van en de kleur Eber gebruikt 9 dispensers. Vervolgens heb ik in Machine 3, 5 extra dispensers ingedeeld voor het doen van rode kleuringen. De rode kleur ae heb ik ingedeeld. Ook ah was een rode kleur, namelijk de bekende kappa/lambda. Dit is de enige kleur die zelf twee dispensers nodig heeft. De oplossing hiervoor was om deze kleur in twee kleuren op te delen, namelijk ah1 en ah2. Deze kleuren zijn ook in Machine 3 ingedeeld. Ag en af zijn ook rood en dus komen ook deze kleuren in Machine 3. Hierna heb ik alle kleuren van klein naar groot gesorteerd. In Machine 1, 2 en 4 was

het aantal dispenserplekken kleiner dan het aantal coupesplekken en dus wordt de kleur met het grootste aantal coupes in Machine 1 geplaatst. Dit zorgt er direct voor dat ook hier het aantal dispenserplekken kleiner of gelijk is aan het aantal coupesplekken. Zo volgen ook Machine 2 en Machine 4. Nu is er voor gezorgd dat in alle machines het aantal dispenserplekken groter is dan het aantal coupesplekken. Het volgende kan nu herhaald worden totdat de machines ingedeeld zijn: Neem de kleur met de meeste coupes en plaats deze in de machine met het grootste aantal coupesplekken. Het resultaat is te zien in bovenstaand schema.

## 6 Conclusies

De subtitel van dit rapport luidt: “Algoritme ontwikkeling voor het optimaal vullen van immunokleuring machines bij de afdeling Pathologie”. Ik ben in de veronderstelling dat ik een zeer degelijk algoritme gemaakt heb die goed uit de verf komt. Het algoritme is getest met gebruikelijke/werkelijke waarden en laat als resultaat een prima indeling van de machines zien.

Het algoritme, zoals deze in een computerapplicatie moet komen, lijkt mij voor iedere programmeur begrijpelijk. Het is geschreven in pseudotaal om het zo makkelijk mogelijk voor de programmeur te maken. Zo hoeft hij namelijk alleen zijn eigen programmeertaal te kennen.

Er bestaat natuurlijk altijd een kans dat een machine niet optimaal ingedeeld wordt. Als er bijvoorbeeld een kleur met 29 coupes in een machine ingedeeld moet worden en er geen kleuren zijn met slechts 1 coupe, dan ontstaat er al een probleem. Er zijn alleen van deze voorbeelden te bedenken met ongebruikelijke waarden! Met gebruikelijke waarden zal het algoritme altijd een optimale indeling geven.

Als Machine 4 slechts enkele Dual Sish kleuringen moet verrichten en er (bijna) geen groot aantal coupes van één kleur gekleurd moet worden, dan is de kans is het grootst dat een machine niet vol ingedeeld kan worden met gebruikelijke waarden. Zo blijft het aantal dispenserplekken in Machine 4 namelijk kleiner dan het aantal coupesplekken en bereikt het aantal dispenserplekken dus eerder zijn limiet van 35. Dit is wel een voorbeeld met gebruikelijke waarden, maar het is natuurlijk niet gebruikelijk dat er (bijna) geen kleuren zijn met meerdere coupes.

Al met al kan ik zeggen dat het algoritme inhoudelijk gezien zeer geslaagd is! Echter heb ik niet kunnen voldoen aan alle eisen voor de gewenste situatie. Als mijn algoritme in een computerapplicatie opgenomen wordt, zullen de Pathologen namelijk zelf moeten tellen hoeveel coupes ze van iedere kleur hebben. Dit moeten ze dan in het programma typen en vervolgens kunnen ze door een druk op de knop de indeling aflezen van het beeldscherm. Het

tellen van de coupes is echter best veel werk. Een oplossing hiervoor is het scannen van de coupes. Op iedere coupe is al een streepjescode aanwezig die afgelezen kan worden door de machines. Als de Pathologen met behulp van een externe scanner de coupes zouden kunnen inlezen in het programma, dan zal dit veel werk besparen. Er valt nog meer tijd te besparen door de coupes niet te scannen, maar al vanaf de eerste aanvraag direct door te laten sturen naar de applicatie. Hierbij zal de applicatie moeten onthouden hoeveel coupes van elke kleur doorgegeven zijn. Dit vergt echter nog wel veel werk, maar is voor de toekomst zeker realiseerbaar!

De Pathologen wouden graag zoveel mogelijk kleuren op één bepaalde machine houden, om zo te voorkomen dat ze over en weer moeten slepen met de dispensers van de kleuren. Er was voor mij niet voldoende goede data aanwezig om na te kunnen gaan welke kleuren dagelijks of bijna dagelijks uitgevoerd worden. Ik ben dus ook niet nagegaan op welke machine deze kleuren die (bijna) dagelijks voorkomen het beste ingedeeld kunnen worden. De afdeling Pathologie kan deze wens zelf eventueel uitvoeren door enkele kleuren zelf in te delen in de machines en vervolgens het aantal dispenserplekken en coupesplekken van de betreffende machine dagelijks aan te passen in de computerapplicatie. Deze mogelijkheid is opengelaten. Ook als met menselijk denkwerk een of meerdere machines beter met de hand ingedeeld kunnen worden, blijft dit mogelijk. Er bestaat namelijk de mogelijkheid om in de applicatie aan te geven dat een of meerdere machines niet ingedeeld hoeven te worden! Met een aantal wensen is dus nog geen rekening gehouden. Dit zou een mooie vervolgpodracht zijn op het algoritme dat ik nu heb ontwikkeld. Er moet altijd ergens begonnen worden en ik denk dat ik een behoorlijk goed begin gemaakt heb en daarbij deels rekening heb gehouden met alle wensen van de afdeling Pathologie van het UMCG.





## 7 Aanbevelingen

Om een goed begin te maken zou ik mijn algoritme laten verwerken in een computerapplicatie door een programmeur. Deze applicatie kan dan getest worden op bruikbaarheid en zal verder uitgewerkt kunnen worden. De programmeur kan ik aanbevelen om dit rapport goed door te nemen om zo op de hoogte te zijn van alle randvoorwaarden. Door het rapport te lezen zal de programmeur mijn algoritme begrijpen en daardoor beter kunnen verwerken tot een applicatie.

De programmeur kan ik de tip geven om rekening te houden met het toevoegen van nieuwe kleuren. Het kan namelijk zo zijn dat de machines nieuwe kleuren moeten verrichten. Deze zullen dan eerst in het programma gezet moeten worden door de programmeur, maar het is natuurlijk handiger als de werknemers van de afdeling Pathologie zelf gemakkelijk kleuren toe kunnen voegen. Omdat de Pathologen voor mijn algoritme de kleuren handmatig moeten tellen, blijft het veel werk voor hen. Door in de toekomst een persoon te zoeken die een uitdaging ziet om de computerapplicatie verder te ontwikkelen, is het scannen van de coupes zeker te realiseren. Zo wordt voorkomen dat de Pathologen handmatig de coupes moeten tellen. Wellicht is het ook mogelijk om direct vanaf de aanvraag het aantal coupes door te sturen naar de applicatie, waarbij de applicatie bij moet houden hoe veel van welke kleur gedaan moeten worden. Hier zal verder onderzoek naar gedaan moeten worden.

Ik denk dat er een mogelijkheid bestaat om bepaalde kleuren in een vaste machine te plaatsen, om zo te voorkomen dat de werknemers de dispensers vanuit de ene machine in de andere machine moeten laden. Op deze manier kunnen de coupes die deze kleuringen moet krijgen namelijk doorgeladen worden terwijl de machine draait. Als de machine klaar is met een coupe, dan kan deze uit de machine gehaald worden en kan er een andere coupe geplaatst worden. Natuurlijk moet de dispenser die bij deze

coupe hoort dan al in de machine zitten! Ook in dit geval zal nader onderzoek nodig zijn. Het vergt namelijk nog wel wat werk om uit te zoeken voor welke kleuren en welke machines deze mogelijkheid bestaat.

Als de Pathologen denken zelf beter een machine in te kunnen delen, om wat voor reden dan ook, dan is dit mogelijk door de betreffende machine uit te schakelen in de applicatie. Het kan bijvoorbeeld zo zijn dat een Patholoog één machine in wil delen met kleuringen die allemaal erg veel tijd in beslag nemen. Deze machine kan dan blijven draaien en de andere machines kunnen een extra keer draaien.

In de verre toekomst is het misschien zelfs mogelijk om een goede computerapplicatie samen te voegen met de gebruikersinterface van de machines. Op deze manier zou ieder bedrijf dat dezelfde machines heeft de applicatie kunnen gebruiken!



## 8 Evaluatie

Mijn afstudeeropdracht heb ik zeer interessant bevonden. Het was een zeer competitieve afstudeeropdracht waar ik veel van geleerd heb. Ik ben in de veronderstelling dat ik een zeer goed eindresultaat geboekt heb waar in de toekomst veel op voortgeborduurd kan worden.

Onderstaand in paragraaf 8.1 een procesevaluatie; de evaluatie van de weg naar het eindproduct.

In paragraaf 8.2 is de productevaluatie opgenomen; de evaluatie van het resultaat van het leerproces.

### 8.1 Procesevaluatie

Mijn eerste onderzoeksvoorstel heb ik te vroeg en wellicht te vluchtig geschreven. Ik heb mijzelf niet genoeg tijd gegund om het probleem te verkennen en onder ogen te zien hoeveel tijd het zoeken van een oplossing ongeveer in beslag zou gaan nemen. Dit heeft er voor gezorgd dat ik iets te gemakkelijk dacht over het schrijven van een programma en te gauw tot een praktijkprobleem ben gekomen. In mijn eerste onderzoeksvoorstel heb ik namelijk gezegd dat ik mogelijk een programma zou gaan schrijven in VBScript of VBA (Visual Basic for Applications). Visual Basic is een programmeertaal voor Officepakketten. Hier ben ik niet aan toegekomen omdat het begrijpen van het hele indelingsprobleem nogal wat tijd gevegd heeft. Aangezien het om een complex probleem gaat, is het vinden van een oplossing ook niet gemakkelijk. Het zoeken van een oplossing in de vorm van een algoritme heeft ook veel tijd en energie gekost. Dit alles heeft ervoor gezorgd dat ik zoveel tijd in het zoeken van een geschikte oplossing heb gestoken, dat ik er niet aan toe ben gekomen om mij goed te verdiepen in VBScript. Dat is de reden dat ik geen computerapplicatie gemaakt heb. Toch heb ik aan het begin van mijn afstudeerproject best veel kostbare tijd besteed aan VBA om in te kunnen zien wat er allemaal mee mogelijk was.

Veel vragen stellen en samenwerken was gedurende de hele periode zeer nodig. Allereerst omdat ik geen verstand had van Pathologie en de machines. Daarnaast omdat ik niet alle informatie verkreeg die ik nodig had. Hiermee wil ik zeggen dat ik de informatie die voor een Patholoog heel logisch is uit hem/haar heb moeten weten te krijgen. Zij hebben dit namelijk niet cadeau gegeven aangezien het voor hen te vanzelfsprekend is. Zo kwam ik er bijvoorbeeld achter, nadat ik zelf verzonnen had dat de verschillende soorten kleuringen het beste op verschillende machines gedaan konden worden, dat de Pathologen dit ook doen. Hier kwam ik achter door een keer aanwezig te zijn bij het indelen van de machines. Alle verzamelde informatie zoals de voorwaarden heb ik duidelijk op papier gezet om zo het wiskundige model te vormen. Alvorens een wiskundige oplossing te zoeken heb ik de verzamelde informatie laten controleren door een medewerker van de afdeling Pathologie. Na goedkeuring ben ik op zoek gegaan naar een wiskundige oplossing voor het probleem.

Ik ben nagegaan hoe ik alle voorwaarden in een algoritme krijgen kon. Ieder idee dat bij mij opkwam heb ik in verhaallijn op papier gezet om zo geen tijd te verspillen aan het onnodig omzetten in algoritmetaal. Ideeën opdoen was echter niet altijd even gemakkelijk. Omdat het probleem complex is en op geen manier op bestaande problemen lijkt, is het lastig vergelijkbaar probleem en oplossing in de literatuur te zoeken wat door niet al te veel aanpassingen als (deel)oplossing voor het machine indelingsprobleem kan dienen. In Hoofdstuk 4 en 5 hebt u kunnen lezen hoe ik tot een oplossing ben gekomen en deze tot een praktijkoplossing heb gebracht. Ik heb in het eindproduct voor zover mogelijk rekening gehouden met alle wensen van de medewerkers van de afdeling Pathologie.

## 8.2 Productevaluatie

Mijn zelf bedachte algoritme is het eindproduct. Het algoritme is geschreven in pseudotaal, om te voorkomen dat de programmeur die het algoritme in een programma gaat verwerken meerdere programmeertalen moet beheersen. Nu is het algoritme door een ieder te begrijpen en kan de programmeur het gemakkelijk omzetten naar de taal van de software waarmee hij programmeert.

Ik vond het jammer dat mijn probleem niet te vergelijken was met verscheidene problemen waar al een oplossing uit de praktijk bekend voor is. Het knapsakprobleem vond ik namelijk erg interessant en heb ik mij veel in verdiept. Dan is het jammer dat ik hier uiteindelijk niets mee heb kunnen doen. Toch zal dit in de praktijk vaker voorkomen. Ik kan een dergelijke oplossing namelijk niet eerder uitsluiten dan dat ik mij er in verdiept heb.

Over het definitieve algoritme ben ik zeer positief. Het is een goed doordacht algoritme geworden. Door continue iets te veranderen aan het algoritme is het beter tot zijn recht gekomen. Hierdoor heb ik namelijk goed na moeten gaan of de veranderingen het algoritme ook verbeterden en waarom. Meestal door zelf te proberen en een enkele keer met hulp van buitenaf. Door een verandering toe te passen zette ik mijzelf weer aan het denken. Zo kwam ik tot nieuwe ideeën die het uiteindelijke algoritme gemaakt hebben zoals het nu is.

Helaas heb ik niet aan alle wensen kunnen voldoen, maar heb ik hier wel rekening mee gehouden. Doordat de machines in het programma uitschakelbaar zullen zijn en de indeling van de machines aan te passen is, zullen veel wensen alsnog handmatig gedaan kunnen worden.

## Literatuur

### Schriftelijke Bronnen

- a. John Walkenbach – Excel 2003 Bible – Part VI Programming Excel with VBA
- b. Robert L. McDonald – An introduction to VBA in Excel
- c. John Clark Craig; Jeff Webb – Microsoft Visual Basic 6.0: Tips, tools en technieken
- d. Theo Peek – Het gebruik van Visual Basic for Applications in Microsoft Excel
- e. G. Buijnes – Basiscursus Programmeren in Office '97
- f. M. Childs – VBScript kort en krachtig
- g. L.C.M. Kallenberg, Universiteit Leiden – Besliskunde 3 2008 – Hoofdstuk 2 Knapzakprobleem(digitale bron nr. 6)
- h. Aris, Rutherford – Discrete dynamic programming
- i. M.L.A. Cremers – Dynamic and stochastic planning problems with online decision making: a novel class of models
- j. D. Ghosh and B. Goldengorin – The binary knapsack problem: Solutions with guaranteed quality
- k. J. J. M. Geurts – Wiskunde en besliskunde 25 jaar later
- l. A.R. van der Burg – De beste beslissing nemen: toepassingen der besliskunde (Operations Research) in het bedrijfsleven
- m. Roel Grit – Projectmanagement – Hoofdstuk 9 Een rapport schrijven

### Digitale bronnen

1. <http://www.ventanamed.com> Informatie over The Benchmark® Ultra machines
2. <http://nl.wikipedia.org/wiki/Immuunhistochemie> Beschrijving van Immuunhistochemie
3. <http://nl.wikipedia.org/wiki/Knapzakprobleem> Nederlandse uitleg van het Knapzakprobleem

4. <http://www.umcg.nl/> Website van de werkgever
5. <http://www.functionx.com/vbaexcel/index.htm> Hulp bij VBA voor Office Excel
6. <http://www.math.leidenuniv.nl/~kallenberg/> Verwijzing naar schriftelijke bron f.
7. <http://www.win.tue.nl/~wscor/OW/2V300/H4.pdf> The Knapsack Problem

## Bijlage I Lijst van Immunokleuringen

1C2	<i>I-view</i>	CD 22		D2-40	
A1 act.		CD 23		DBA-44	
a1AT		CD 30		Desmine	
Act. PAN		CD 31		Dual Sish	<i>Dual Sish</i>
Act. SMA		CD 34		Eber	<i>Eber</i>
ACTH		CD 34 crista		EBV LMP-1	
AFP		CD 35		E-Cadherine	
Alfa-Synucleine	<i>I-view</i>	CD 43		EGFR	
ALK		CD 45		EMA	
Amyloid AA		CD 45RO		ER	
Amyloid beta	<i>I-view</i>	CD 56		F VIII	
Amyloid SAP		CD 57		Fibrinogeen	
Amyloid transt.		CD 61		FSH	
BCL2		CD 68 (KP1)		Gastrine	
BCL6		CD 68 (PGM1)		GCDFFP-15	
BerEp4		CD 79 A		GFAP	
Beta-catenine		CD 99		Glucagon	
Borrelia/treponema		CD117		Glycophorine	
C4d		CD138		Groeihormoon	
Calcitonine		CEA		HBC	
Caldesmon		Chromogranine		HBS	
Calponine		CK 1/5/10/14		HCG beta	
Calretinine		CK14		Helicobacter	
CD 1 A		CK 19		Her2/Neu	
CD 2		CK 20		HHV8	
CD 3		CK 5/6		HMB45	
CD 4	<i>Rood</i>	CK 7		HPL	
CD 5		CK 8/18		HSV1	
CD 8		CK AE N 1,2		I9G4	
CD 10		CMV		Ig K.br/L.rd	<i>Rood + Ultra</i>
CD 15		Collag. IV		Ig kappa	

CD 20		Connexine-43		Ig lambda	
CD 21		Cycline D1		IgA	
IgD		Myoglobuline		Serotonine	
IgG		Neurofilament		Somatostatine	
IgM		NPM		SQSTM1 (p62)	<i>I-view</i>
Inhibine		NSE		SV40	
Ini-1		OC-125		Synaptophysine	
Insuline		P16		TAU	
KI-67		P53		TDT	
LH		P63		Thyreoglobuline	
Lysozym		Pax5		TIA-1	
MDM-2		Plakoglobine		TLE-1	
Melan A	<i>Soms Rood</i>	PLAP		Toxoplasma	
MLH 1		PMS-2		Tryptase	
MOC-31		PPP		TTF-1	
MPO		PR		Ubiquitine	
MSH 2		Prolactine		VC38	
MSH 6		PS7		Vimentine	
MUM1/IRF4		PSA		WT-1	
Myf 4		PZF		ZAP-70	
Myo D1		S-100	<i>Soms Rood</i>		

Bovenstaand de tabel met alle immunokleuringen van dit moment op alfabetische volgorde.

Er kunnen in de toekomst altijd meer kleuringen bij komen.

De kleuringen waar geen soort kleur achter staat zijn Ultra-view.



## Bijlage 2 Elke machine heeft een vaste kleuring

We hebben 4 machines met elk hun vaste kleuringen:

<b>Machine 1</b>		<b>Machine 2</b>		<b>Machine 3</b>		<b>Machine 4</b>	
I-view	8	Eber	9	Rood	5	Dual Sish	17
Ultra-view	8	Ultra-view	8	Ultra-view	8	Ultra-view	8

Onder de volgende voorwaarden worden er x aantal dispenserplekken bezet:

Hematoxyline	2 (wordt nu weer handmatig gedaan)
Ultra-view	8
I-view/Eber/Rood/Dual Sish	X (zie bovenstaande tabel)
Elk verschillend soort kleuring	1 (antilichaam, ongeacht aantal coupes)

### Bijlage 3 Aanpassen dispenserplekken onder voorwaarde

Elke machine heeft 27 overgebleven en dus beschikbare plekken voor Dispensers. Er zijn namelijk al 8 bezet door Ultraview. Onder de volgende voorwaarden wordt het aantal beschikbare plekken voor dispensers voor de betreffende machine aangepast:

<b>Voorwaarde: Zijn er...</b>	<b>Als "Ja", dan</b>
I-view kleuringen?	Machine 1 heeft 19 plekken voor dispensers
Eber kleuringen?	Machine 2 heeft 18 plekken voor dispensers
Rode kleuringen?	Machine 3 heeft 22 plekken voor dispensers
Dual-Sish kleuringen?	Machine 4 heeft 10 plekken voor dispensers

## Bijlage 4 Schema Roosterprobleem

In onderstaand schema heb ik duidelijk proberen te maken waarom ik het machine indelingsprobleem niet kon vertalen naar een Roosterprobleem.

### *Voordelen*

Ik kan het overzicht over de 4 machines gemakkelijk bewaren. Te zien is dat Machine 4 op 5 dispensers na vol zit.  
Ik kan een maximum van 35 stellen voor het aantal dispensers per machine.  
Ik heb de soorten kleuringen in het schema kunnen verwerken.

### *Nadeel*

Ik kan geen limiet van 30 stellen aan het aantal glaasjes per machine.  
Ook is niet in één oogopslag duidelijk hoeveel glaasjes er momenteel in de machines geplaatst zijn.

### *Conclusie*

Van mijn probleem kan ik geen roosterprobleem maken. Via deze weg valt er dus geen algoritme te ontwikkelen.

	8	13	16	17	25	28	30	35
<b>Machine 1</b>	Ultra View	I-view? Ja/Nee		9 Amyloid beta				
<b>Machine 2</b>	Ultra View	Eber? Ja/Nee		11 antilichaam Eber				
<b>Machine 3</b>	Ultra View	Rood? Ja/Nee	4 CD4					
<b>Machine 4</b>	Ultra View	Dual-Sish? Ja/Nee				5 Dual Sish		

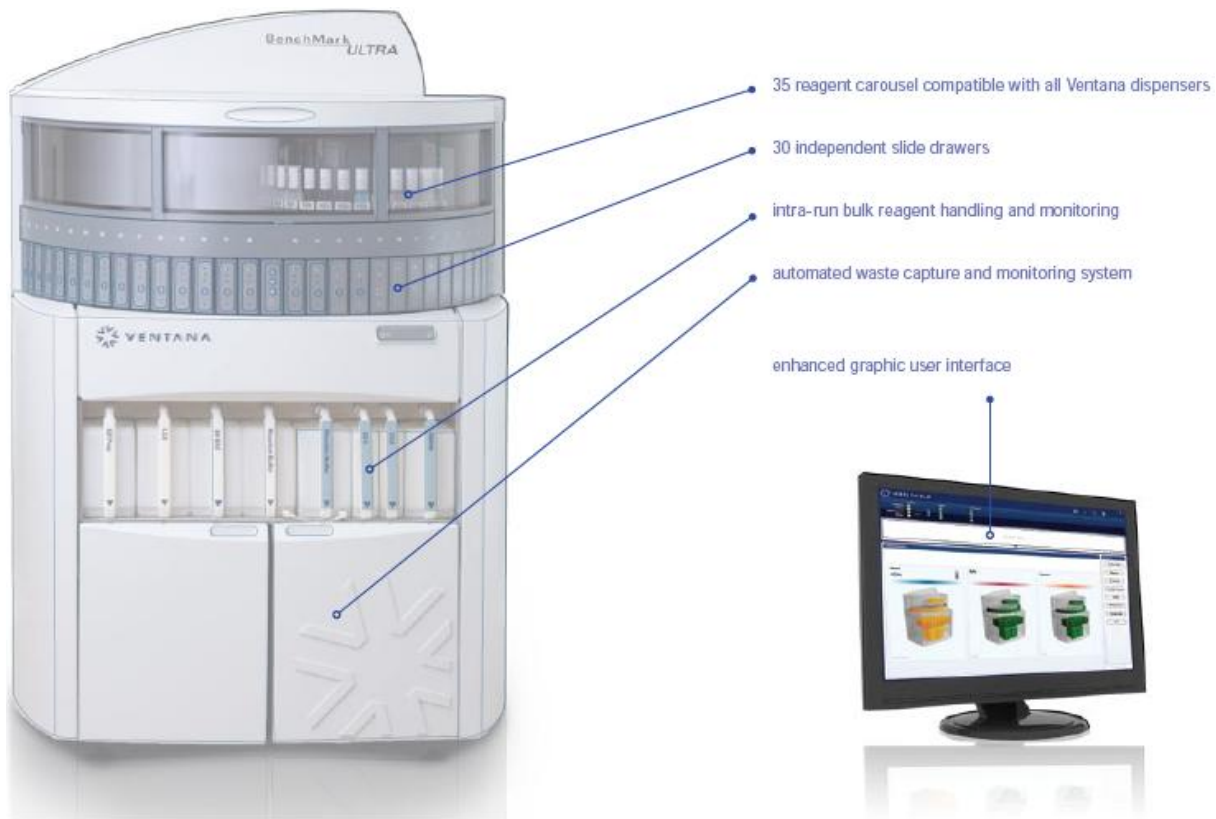
## Bijlage 5 The Benchmark® Ultra

Om uw beeld van de kleuringmachines iets duidelijker te maken heb ik onderstaand een aantal afbeeldingen.

Afbeelding 1 is het totaalplaatje van The Benchmark® Ultra. In het bovenste deel van de machine kunt u door het glas kijken. Hier ziet u de dispensers (totaal 35 posities). De grijze vakjes daar onder kunnen stuk voor stuk open. Er zijn 30 van deze vakjes, elk voor één coupe. De onderste 8 vaten bevatten stoffen die de machine gebruikt. Hier heb ik geen rekening mee hoeven houden in mijn algoritme. Daaronder vangt de machine de afvalstoffen op.

Afbeelding 2 is het user-interface, zoals die ook aanwezig is op de afdeling Pathologie op het UMCG. Alles qua instellen van de machines gebeurt via deze computer.

36



Afbeelding 1

Afbeelding 2

Op Afbeelding 3 wordt een coupe in de machine geplaatst. Op de coupe is een streepjescode aanwezig, zodat de machine weet welke kleur het weefsel moet krijgen. Het weefsel bevindt zich op het glaasje, maar is zo dun dat het bijna niet zichtbaar is.



*Afbeelding 3*