# Proceedings of the thirty-sixth European Study Group with Industry (ESGI36) : Eindhoven, The Netherlands, 15-19 November 1999

**EINDHOVEN UNIVERSITY OF TECHNOLOGY**
Department of Mathematics and Computing Science

# Proceedings of the Thirty-sixth European Study Group with Industry (ESGI36)

# Verslag van de Studiegroep Wiskunde met de Industrie '99 (SWI'99)

J. Molenaar (editor)

Eindhoven, The Netherlands, 15-19 November 1999

EUT Report 00-WSK-01
Eindhoven, July 2000

Department of Mathematics and Computing Science
Eindhoven University of Technology
P.O. Box 513
5600 MB  Eindhoven, The Netherlands
ISSN: 0167-9708
Coden: TUEEDE

$36^{th}$ European Study Group with Industry
15-19 November, 1999

# Mathematicians can solve your problems

Mathematics is a noble science that has existed for centuries. But it can also be used as a tool to solve practical problems. Although some problems may sound uncomplicated when expressed in words, formulating them in a mathematical way may be no simple matter. In the Netherlands we are lucky enough to have an enthusiastic group of mathematicians who like to meet the challenges of the real world. They took part in the 36th European Study Group with Industry in Eindhoven in 1999. The participants were confronted with wind tunnels and oil wells, with transistors, memories and traffic planners, with desk tops and paint. All these items have nothing in common but the possibility of developing a mathematical model.

During a week a group of Dutch and foreign mathematicians applied their joint ingenuity to sometimes sticky cases of practical problems brought to them by several companies: NLR, Schlumberger, Philips, KPN, Ericsson, Trespa and Akzo Nobel. This did not dampen their spirits in any way. On the contrary, they enjoyed the common efforts: the atmosphere was excellent. For mathematicians unsolved problems are a joy because it is a challenge to help others with things that are out of reach to them. During the week in Eindhoven close contact existed between the problem owners and the Study Group in order to clarify all necessary questions about the context of the cases. At the end of the week every subgroup of the Study Group presented its own models and suggestions for a solution. These are condensed or maybe grown out into the manuscript you find in your hands.

The Dutch research program Wiskunde Toegepast (Mathematics Applied) is proud that this second Study Group in the Netherlands was a success. The financial support was well spent as it showed the world the potential of mathematics as a practical problem solver. The aim of Wiskunde Toegepast is to develop mathematics as a science and a practical tool. Wiskunde Toegepast is a joint program of the Technology Foundation STW and the Council for Physical Sciences of NWO.

Marijke de Jong,
Secretary of the program "Wiskunde Toegepast".

$36^{th}$ European Study Group with Industry
15-19 November, 1999

# Table of Contents

**Contributions:**

36$^{th}$ European Study Group with Industry
15-19 November, 1999

# Short summaries of the problems dealt with in ESGI36

**Akzo Nobel**: *Mixing colors*

Suppose you keep a limited number of basic colors in stock, and whenever a customer demands paint of a certain color, you try to create this color by mixing paints from your stock. The problem is to decide which basic colors you need, such that you are able to create acceptable approximations to the colors that are usually in demand.

**Philips NatLab**: *Compact models for high-voltage MOS devices*

An IC (integrated circuit) consists of many (connected) semiconductor devices (transistors). The electrical behaviour of a single transistor can be calculated numerically. The electrical behaviour of the IC as a whole is calculated by coupling the functioning of the individual transistors. In this process it is too timecomsuming to use the numerical evaluations of the transistors. So-called compact models are needed which provide explicit analytical expressions for the relations between the voltages and currents of a transistor. The problem is to derive a reliable compact model for a given transistor.

**Trespa International**: *Control of panel manufacturing*

Trespa is a leading manufacturer of decorative panels. These panels consist of a cured phenol/formaldehyde resin reinforced with wood or cellulose fibres. Due to the high natural fibre content, the panels (sometimes) warp under moisture variations of the environment. The goal of the project is to develop a model with which panel warp can be predicted. Such a model will probably include the description of the thermo-chemical reactions during panel formation and the mechanical deformation due to internal stresses. Relevant parameters of the processes involved are available.

**NLR**: *Determining position and orientation of a windtunnel model from a single optical picture*

Traditionally, measuring the flow and determining the model's position are treated as different tasks, and therefore performed with the use of different measurement systems. However, if the method flow is determined optically with a camera, the position of the model could in principle be determined at the same time, together with the flow. The question is to determine the model position and the angles of orientation quickly, efficiently, and accurately from a single picture if the geometry of the model is fully known.

**KPN Research**: *Efficient use of memory in WWW-caches*

A key poblem of the WWW is its lack of scalability. Improvement of the performance is reached via caching: frequently accessed Web documents are stored not at one place, but at several, carefully selected links. Given the fact that cache memory is relatively expensive compared to disk space, the question arises how to properly engineer caching in the network. The specific problem is to allocate amounts of cache memory in a specific network structure with known traffic characteristics, taking into account budget limitations.

**Schlumberger**: *Estimation of the thickness of wall layers in inclined slots*

After drilling of an oil well, a steel casing or liner is run into the bottom hole. Between the steel tube and the rock formation a more or less annular gap is left. The problem concerns the displacement of the drilling mud from this gap. This is done by pumping a sequence of fluids down the inside of the tube from surface and returning up towards the surface in the gap. The problem is that sometimes a residual layer of the displaced fluid sticks to the wall. This gives rise to a mud layer left in the gap which prohibits the complete filling of the gap with cement. This effect may reduce the productivity of the well considerably. Some models to describe these effects have already been developed for idealized slots. Models for realistic geometries are needed.

**Ericsson**: *Route-information from a central routeplanner*

In view of the increasing number of traffic jams on the Dutch roads, the provision of information on which routes cause the least delays becomes more and more important. It is, of course, necessary that a routeplanner uses the most recent information on traffic densities at the different roads, because the traffic situation on the roads constantly changes. In order to achieve this, road users could pass on information about their staring point, destination and planned route to a central server. Using this information, the central server can estimate future traffic streams and hence predict traffic jams. The more road users passing on their information, the more valuable the system with a central routeplanner will become. Question is how many road users should participate in order to get a significant improvement of prediction of traffic jams.

# Author information

Problem from **AKZO-Nobel**:

R. Canogar, Universidad Nacional de Educación a Distancia, Madrid, Spain, rcanogar@math.uned.es
R. Pendavingh, Eindhoven University of Technology, rudi@win.tue.nl

Problem from **Philips NatLab**:

F.P.H. van Beckum, University of Twente, frits@math.utwente.nl
J. Boersma, University of Eindhoven, j.boersma@tue.nl
L.C.G.J.M. Habets, University of Eindhoven, luch@win.tue.nl
G. Meinsma, University of Twente, g.meinsma@math.utwente.nl
J. Molenaar, University of Eindhoven, j.molenaar1@tue.nl
W.H.A. Schilders, Philips Research/University of Eindhoven, schildr@natlab.research.philips.com
A.A.F. van de Ven, University of Eindhoven, fonsvdv@win.tue.nl

Problem from **Trespa**:

D. Chandra, University of Eindhoven, d.chandra@tue.nl
H.J.J. Gramberg, University of Eindhoven, h.j.j.gramberg@tue.nl
T. Ivashkova, University of Eindhoven, tatsiana@win.tue.nl
W.R. Smith, University of Eindhoven, warren@win.tue.nl
A. Suryanto, University of Twente, a.suryanto@math.utwente.nl
J.H.M. ten Thije Boonkkamp, University of Eindhoven, tenthije@win.tue.nl
T. Ulicevic, University of Eindhoven, tanjau@win.tue.nl
J.C.J. Verhoeven, University of Eindhoven, keesverh@win.tue.nl

Problem from **NLR**:

R. Stoffer, University of Twente, r.stoffer@el.utwente.nl
C. Stolk, University of Utrecht, stolk@math.uu.nl
S.W. Rienstra, University of Eindhoven, S.W.Rienstra@tue.nl
J.K.M. Jansen, University of Eindhoven, J.K.M.Jansen@tue.nl

Problem from **KPN Research**:

W.J. Grootjans, University of Eindhoven, wj@mediaport.org
M. Hochstenbach, University of Utrecht, hochsten@math.uu.nl
J. Hurink, University of Twente, hurink@math.utwente.nl
W. Kern, University of Twente, kern@math.utwente.nl
M. Luczak, University of Oxford, luczak@maths.ox.ac.uk
Q. Puite, University of Utrecht, puite@math.uu.nl
J.A.C. Resing, University of Eindhoven, resing@win.tue.nl
F.C.R. Spieksma, University of Maastricht, spieksma@math.unimaas.nl

Problem from **Schlumberger**:

B.W. v.d. Fliert, University of Twente, fliert@math.utwente.nl
J.B. v.d. Berg, University of Leiden, gvdberg@math.leidenuniv.nl

Problem from **Ericsson**:

G. Hek, University of Amsterdam, ghek@science.uva.nl
G. Lunter, University of Groningen, gerton@math.rug.nl
J.A.M. Schreuder, University of Twente, j.a.m.schreuder@math.utwente.nl
J. White, University of Oxford, white@maths.ox.ac.uk

# Color Matching

R. Canogar and R. Pendavingh

**Abstract**

We consider the problem of creating paint of a certain target color by mixing colorants. Although a large number of colorants is available, in practice it is only allowed to use a limited number. We focus on the problem of selecting the right subset of colorants.

**Keywords**

Linear programming, Paint mixing, Kubelka-Munk model, Greedy algorithm.

## 1 Introduction

Consider the following problem. A car enters a garage for repair. The paint layer of the car has been damaged. We want to repair the damage without completely repainting the whole car. To remove every trace of damage, we will locally apply paint of a color very similar to the color of the paint that is already on the car. The problem is, where do we get paint of the right color? Usually, such paint will not be available ready-made. We need to create it ourselves by the familiar process of mixing colorants.

In the next section we describe a simple model due to Kubelka and Munk that allows us to predict the color of a mixture of colorants, given a recipe specifying that colorant $i$ is used in relative proportion $c_i$.

Using this model it is possible, given a target color and a set of colorants, to compute a recipe that produces a best approximation to the target color.

This seems to solve our initial problem completely (assuming that the Kubelka-Munk model is accurate), but there is a catch. For several reasons, we do not want to use too many colorants in a recipe, not more than $k$ say, whereas there are many more colorants available, say $n$. We need to select, given our target color, a good set of $k$ colorants to use in our recipe. In fact, we are interested in a couple of $k$-sets that produce a good approximation to the target color.

Of course, we could compute a recipe for each $k$-set of colorants, and then decide which $k$-sets produce the best approximations to our target color. Since computing one recipe is already nontrivial, and $n$ over $k$ will be an exceedingly large number, this takes too much time. Moreover, many of the $k$-sets will only produce very poor approximations to the target color ($k$ shades of blue will never make a good red), and it seems wasteful to precisely compute many recipes when only a few good ones are needed.

In this paper, we will explain an approach that could be used to wield out bad $k$-sets without computing many recipes.

## 2 The Kubelka-Munk model

For a given painted surface and a wavelength $\lambda$, the *reflectance* $R(\lambda)$ is defined as the proportion of light of wavelength $\lambda$ that is reflected by the paint layer. The *color* of the surface is determined by the reflectance values of light in the visible spectrum.

Colorants have two parameters, the *absorption* $K(\lambda)$, and the *scattering* $S(\lambda)$, both depending on the wavelength $\lambda$. We may assume that we know both parameters for each of our colorants.

The Kabulka-Munk model predicts that a completely hiding paint layer will satisfy the following relation between the reflectance and the parameters of the colorant, for each wavelength $\lambda$:

$$\frac{K(\lambda)}{S(\lambda)} = \frac{(1 - R(\lambda))^2}{2R(\lambda)}. \tag{1}$$

Moreover, when we mix colorants $1, \ldots, k$ in relative proportions $c_i$ (so $\sum_i c_i = 1$ and $c_i \geq 0$) we have

$$K(\lambda) = \sum c_i K_i(\lambda), \tag{2}$$

and

$$S(\lambda) = \sum c_i S_i(\lambda), \tag{3}$$

where $K_i, S_i$ are the coefficients of colorant $i$ and $K, S$ are the coefficients of the mixture.

In practice, we consider only a finite number of frequencies $\lambda_1, \ldots, \lambda_l$ adequately representing the visible spectrum. That is, we measure the reflectance values $R_t(\lambda_1), \ldots, R_t(\lambda_l)$ of our target color. A mixture of colorants is considered a very good approximation if it has the same reflectance values at wavelength $\lambda_1, \ldots, \lambda_l$. Let us say that a set of colorants $I$ is *very good* if there is a recipe using only colorants in $I$ that gives a very good approximation of the target color.

From (1) – (3) we derive that a set $I$ is very good if and only if there exist $c_i \geq 0$ such that:

$$\frac{\sum_{i \in I} c_i K_i(\lambda_j)}{\sum_{i \in I} c_i S_i(\lambda_j)} = \frac{(1 - R_t(\lambda_j))^2}{2R_t(\lambda_j)}, j = 1, \ldots, l. \tag{4}$$

Since the human eye does not perceive color with the precision of a spectrometer, a 'very good' set of colorants is in fact more than we need. But let us concentrate on very good sets of colorants for now.

## 3 A geometrical view

Rewriting (4) we obtain the following. The set $I$ is very good if there exist $c_i \geq 0$ such that

$$\sum_{i \in I} c_i w_i = 0, \tag{5}$$

where the $w_i$ are vectors in $\mathbb{R}^l$ whose $j$-th coordinate is defined by

$$(w_i)_j := K_i(\lambda_j) - \frac{(1 - R_t(\lambda_j))^2}{2R_t(\lambda_j)} S_i(\lambda_j). \tag{6}$$

Note that the vector $w_i$ is completely determined by the parameters of the colorant $i$ and the reflection values of the target color.

Equation 5 has a simple geometric interpretation: it states that $I$ is a very good set if and only if the origin is in the convex hull of $\{w_i \mid i \in I\}$.

Now remember that our goal is to limit the size of the set of colorants used in the recipe, i.e. limit the cardinality of $I$ by $k$. So when we look for very good sets of colorants, we are faced with the following geometrical problem:

> Given a set of vectors in $\mathbb{R}^l$, select a subset of at most $k$ vectors whose convex hull contains the origin.

If the vectors $w_1, \ldots, w_n$ are in general position, a subset of these vectors containing the origin in its convex hull will have at least $l + 1$ elements. In other words, there are no very good sets of cardinality $\leq l$. This is a problem, since $k$ is usually less than $l$ in our application, and there is no reason why the vectors shouldn't be in general position.

It is time to use the fact that the human eye can be fooled, and determine when a set of colorants is good enough.

# 4 The eye

On the retina of the human eye there are light sensors of three types, each type maximally sensitive to light of a distinct wavelength. Light entering the eye will stimulate sensor of type $t$ ($t = 1, 2, 3$) proportional to

$$z_t := \sum_j A(\lambda_j) a_{jt} \tag{7}$$

where $A(\lambda)$ is the absolute intensity of light of wavelength $\lambda_j$ entering the eye and $a_j$ is the relative sensitivity of type $t$ to light of wavelength $\lambda_j$. The vector $z := (z_1, z_2, z_3)$ is all the information the brain gets from the entering light: thus our color sense is essentially 3-dimensional, and there is a linear map $Z : (A(\lambda_j))_j \mapsto z$.

For any fixed $z$, the eye is unable to distinguish between any two kinds of light with absolute intensity vectors in $Z^{-1}(z)$.

The color of light emitting from a painted surface depends on both the reflection values $R(\lambda)$ of the paint layer and the environmental light illuminating the surface. By definition of $R$, we have $A_{out}(\lambda_j) = R(\lambda_j)A_{in}(\lambda_j)$ for each wavelength $\lambda_j$. So given a certain kind of environmental light $e$, we have a linear map $Y_e : (R(\lambda_j))_j \mapsto (A_{out}(\lambda_j))_j$. This somewhat enhances the ability of the eye to distinguish paint colors. Two paint layers, with reflection vectors $r_1, r_2$ can appear to the eye to have the same color in one kind of environmental light ($Z(Y_1(r_1)) = Z(Y_1(r_2))$), but can be seen to have a different color under another kind of light ($Z(Y_2(r_1)) \neq Z(Y_2(r_2))$). This phenomenon is known as *metamerism.*

In practice, a car is not looked at under every possible kind of light, and this makes our job somewhat easier. We may assume that the repaired car will only be scrutinized in a very limited set of environments, say in daylight and in the light that is usually emitted by street lamps. This means that if the paint layer on the car has reflection vector $r_t := (R_t(\lambda_1), \ldots, R_t(\lambda_l))$, it is satisfactory if we create paint with reflection vector $r$ such that

$$Z(Y_{daylight}(r)) = Z(Y_{daylight}(r_t)), \tag{8}$$

and

$$Z(Y_{streetlight}(r)) = Z(Y_{streetlight}(r_t)) \tag{9}$$

This procuces a set $\mathcal{R} \subseteq I\!R^l$ of reflection vectors that can be safely substituted for the target reflection vector $r_t$. The solution set of (8) is an affine subspace of $I\!R^l$ but note that reflection values should be between 0 and 1. We may even want to restrict ourselves to $R(\lambda_j)$ between $R_t(\lambda_j) \pm \epsilon$. In any case, $\mathcal{R}$ will be a convex set.

We will say that set of colorants $I$ is *good enough* if by mixing colorants from $I$ we can create paint with reflection vector $r \in \mathcal{R}$.

From the previous section it follows that $I$ is good enough if and only if there is some reflection vector $r = (R(\lambda_1), \ldots, R(\lambda_l)) \in \mathcal{R}$ such that

the origin lies in the convex hull of $\{w_i^r \mid i \in I\}$,

where

$$(w_i^r)_j := K_i(R(\lambda_j)) - \frac{(1 - R(\lambda_j))^2}{2R(\lambda_j)} S_i(\lambda_j). \tag{10}$$

Thus $w_i = w_i^{r_t}$.

# 5 Two methods

## 5.1 The random hyperplane method

Consider a finite set of vectors $U$ in $I\!R^l$. It is clear that if $U$ is strictly on one side of a (linear) hyperplane $H$, then the convex hull of $U$ does not contain the origin. From geometrical intuition it is also obvious (but not trivial to prove) that the converse holds: namely that if the convex hull of $U$ does not contain the origin, there is some hyperplane $H$ having all of $U$ strictly on one side. Such is the content of Farkas' Lemma (see e.g. [2]):

**Lemma 1 (Farkas)** *Given a finite set of vectors $U \subseteq \mathbb{R}^l$ exactly one of the following statements hold:*

1. *$0$ lies in the convex hull of $U$, and*
2. *there is a vector $d \in \mathbb{R}^l$ such that $(d, u) > 0$ for all $u \in U$.*

By $(.,.)$ we denote the inner product of two vectors.

This can be put to use for our problem in the following way.

Let us first consider 'very good' sets of colorants again. If we take an arbitrary vector $d \in \mathbb{R}^l$, and set $F_d := \{i \mid (d, w_i) > 0\}$, then it follows from (the easy part of) Farkas' Lemma that any set of colorants $I$ with $I \subseteq F_d$ will not be a very good set. The nontrivial part of Farkas' Lemma shows that any set that is not 'very good' has a nonzero chance of being a subset of $F_d$ . We may rapidly construct a multitude of such 'forbidden' sets, by randomly choosing vectors $d_1, \dots, d_N$ from $S^{l-1} := \{d \in \mathbb{R}^l \mid \|d\| = 1\}$. Then we search for $k$-sets $I$ that satisfy $I \setminus F_{d_i} \neq \emptyset$ for all $i = 1, \dots, N$, and provided that $N$ is big enough such an $I$ will very likely be a very good set.

If we are interested in sets that are 'good enough', the problem becomes more subtle. Define for each colorant $i$ the set $W_i := \{w_i^r \mid r \in \mathcal{R}\}$ where $\mathcal{R}$ is the set of safe substitutes for the target reflection vector $r_t$ of the previous section. Given any vector $d \in \mathbb{R}^l$, we put

$$F_d := \{i \mid \min_{w \in W_i} (d, w) > 0\}. \tag{11}$$

Clearly, no subset of $F_d$ will be good enough. It is not true anymore that any set that is not good enough is eliminated this way. Still, we can construct many such forbidden sets $F_d$ each time killing many candidate $k$-sets of colorants.

The minimization problem $\min_{w \in W_i}(d, w)$ is hard in general but

1. we may assume that $\mathcal{R}$ is a polytope, and
2. we can replace $w_i^r$ by its linear approximation around $w_i^{r_t}$ provided that $\|r - r_t\|$ is small,

yielding a polytope $\tilde{W}_i$ approximating $W_i$. We can either solve the minimization problems $\min_{w \in \tilde{W}_i}(w, d)$ for each $d$ or compute the vertices $V_i$ of $\tilde{W}_i$ in advance, and use the fact that $\min_{w \in \tilde{W}_i}(w, d) = \min_{v \in V_i}(v, d)$ for every $d$.

A faster, but more crude method is to replace the condition

$$\min_{w \in W_i} (d, w) > 0$$

in (11) by $(d, w_i) > \epsilon$ for some strategically chosen $\epsilon > 0$. Thus we use the unquantified notion that we can still displace the vectors $w_i$, but only a little. If a set of vectors is far on one side of a linear hyperplane, the chances that a small displacement of these vectors has the origin in its convex hull become very thin.

When a suitable collection of forbidden sets $F_1, \dots, F_N$ has been constructed, it remains to find sets $I$ such that $I \not\subseteq F_i$ for all $i$. Equivalently, we want an $I$ such that $I \cap \overline{F_i} \neq \emptyset$ for all $i$, where $\overline{U}$ denotes the complement of a set. This is known in the literature as a *set covering problem* : the set $I$ needs to 'cover' each $\overline{F_i}$.

Although there is no direct relation to the current problem, the approach described in this section was inspired by the method described in [1].

## 5.2 The greedy algorithm method

The greedy algorithm is a very general method to pick a good $k$-subset $I$ out of an $n$-set $C$. To apply the greedy algorithm we need a measure $f$ of how good a subset is. For the moment we will not specify this function, we only remark that it acts on all subsets of $C$ and its image is a real value. Applied to our problem, this algorithm looks like this:

```
1. $I \leftarrow \emptyset$; $c \leftarrow 0$.
2. choose $i \in C$ such that $f(I \cup \{i\})$ is maximal
3. $I \leftarrow I \cup \{i\}$; $c \leftarrow c + 1$
4. IF $c = k$ THEN RETURN($I$)
5. goto 2.
```

In step 2 we intentionally do not specify that $i \notin I$. In that way $I$ is increased with some element only when this results in an improvement.

The main advantage of this algorithm is that it runs very fast. The speed depends heavily on how fast can we evaluate the function $f$. The drawback is that we may end with a far from optimal subset $I$. We will start very well picking the first elements but the subsequent choices made in step 2 can be very weak. One solution to this problem is to incorporate some flexibility (an integer $m$ will be the measure of flexibility). Now we give a second version of the greedy algorithm with flexibility $m$:

```
1. $I_1 \leftarrow \emptyset; \cdots I_m \leftarrow \emptyset$; $c \leftarrow 0$
2. find $m$ elements $(i_1, j_1), \dots, (i_m, j_m)$ in $C \times \{1, \dots, m\}$ that take the $m$ maximal values (in
   order) of the function $(i,j) \rightarrow f(I_i \cup \{j\})$ and such that for all $1 \leq s,t \leq m$, $I_{i_s} \cup \{j_s\} \neq$
   $I_{i_t} \cup \{j_t\}$.
3. $I_1 \leftarrow I_{i_1} \cup \{j_1\}, \dots, I_m \leftarrow I_{i_m} \cup \{j_m\}$; $c \leftarrow c + 1$
4. IF $c = k$ THEN RETURN($I_1, \dots, I_m$)
5. goto 2.
```

This method will approach the optimum as we increase $m$. It is also interesting to have more than one set of colorants to mix, for example one set of colorants may be more stable under small variations on the concentrations than others.

Now we will look at two different functions $f$ or measures on how good a set of colorants is. With them we will try to get as close as possible to a very good set of colorants and we will not treat the more difficult problem of finding a good enough solution.

### 5.2.1 Minimum distance

Remember from section 3 the geometrical interpretation of equation (5): $I$ is very good if the origin is contained in the convex hull $H_I$ of $\{w_i | i \in I\}$. Also we remarked that this will not happen in general, so our aim is that the convex hull is as close as possible to the origin. Thus the Euclidean distance from the convex hull to the origin is the natural way to judge sets $I$:

$$f(I) = d(0, H_I) = \min\{\|w\| \mid w \in H_I\}. \tag{12}$$

We describe how to calculate this with the following program. By $V_K$, for any $K \subset I$, we mean the affine space generated by $\{w_i\}_{i \in K}$.

```
1. $I_1 \leftarrow I; v_1 \leftarrow 0; i \leftarrow 1$
2. Let $\pi_{I_i}(v_i)$ be the orthogonal projection of $v_i$ onto $V_{I_i}$.
```

```
3. IF exists $J_i \subset I_i$ such that $V_{I_i \setminus J_i}$ is an hyperplane in $V_i$ and separates $\{w_j\}_{j \in J_i}$ from $\pi_{I_i}(v_i)$
   THEN goto 4
   ELSE RETURN($\sqrt{d(v_1, v_2)^2 + \cdots + d(v_{i-1}, v_i)^2}$)
4. $I_{i+1} \leftarrow I_i \setminus J_i$; $v_{i+1} \leftarrow \pi_{I_i}(0)$; $i \leftarrow i + 1$
5. goto 2.
```

### 5.2.2 Angle

Let us suppose for a moment that $k = 2$, this means that we have a collection of points $\{w_i\}_{i \in C}$ and our aim is to pick two points $w_{i_1}, w_{i_2}$ such that the interval between both almost contains the origin. If this is the case then the angle $\widehat{w_{i_1} 0 w_{i_2}}$ should be very close to $\pi$. And it holds that the angle $\widehat{w_{i_1} 0 w_{i_2}}$ is $\pi$ if and only if 0 is contained in the interval between $w_{i_1}$ and $w_{i_2}$. This suggests that the angle might be a good measure. Unfortunately there are very particular cases where this measure is bad. So we will hope that our set of points is general enough and believe that this measure is good.

What happens if $k > 2$? We propose a generalization of angle, namely the solid angle: fraction of the unit sphere overlapped by the cone with the origin as a vertex and generated by the convex hull of our set of points. This number is not easy to calculate unless $k \leq 3$ (for $k = 3$ we have the Gauss-Bonet formula), so the best way to approximate it is by a Montecarlo method. That is, we select random unit vectors uniformly distributed and we count how many lie inside the cone. If we do this for enough vectors we will get a good approximation of the solid angle. For $k > 2$ it is still true that the solid angle of the cone with vertex 0 and generated by $\{w_i\}_{i \in I}$ is $1/2$ of the unit sphere if and only if the convex hull of $\{w_i\}_{i \in I}$ contains the origin.

One further idea is to use as a measure the sum of the angles between all pairs of vectors in $\{w_i\}_{i \in I}$. This works well only if $k$ is small (say $k \leq 6$). For example for $k = 3$ the sum of the angles of all pairs is $2\pi$ if and only if the convex hull $H_I$ contains the origin. For $k \leq 4$ there is still a maximum for the sum, in the case when this maximum is attained then the origin is in $H_I$ but the reciprocal does not hold any more.

## 6 Conclusion

The methods presented in this paper were the result of a week-long brainstorming session. There is nothing final about any of the algorithms we describe. Rather, we show that the hard problem of selecting colorants has a geometrical interpretation that inspires a new kind of strategy to solve the problem.

## References

[1] M. Goemans and D. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. Assoc. Comput. Mach. 42 (1995), no. 6, 1115–1145.

[2] A. Schrijver, *Theory of Linear and Integer Programming*, Wiley-interscience series in discrete mathematics, John Wiley & Sons, Ltd., Chichester, 1986.

# On compact models for high-voltage MOS devices

F.P.H. van Beckum, J. Boersma, L.C.G.J.M. Habets,
G. Meinsma, J. Molenaar, W.H.A. Schilders, A.A.F. van de Ven

**Abstract**

Fast evaluation of integrated circuits(ICs) requires the availability of so-called *compact models*, i.e. simple-to-evaluate relations between the voltages and the currents in the IC-components. In this paper the compact model for a particular IC-part, the *LDMOS device*, is studied. This model consists of coupled submodels, each of which describes a separate part of the LDMOS device. The purpose of the present work is the derivation of the submodel for the transition region of the LDMOS. As a preparation a model for a neighbouring region, the drift region, is derived in full detail. It is shown that the submodels for transition and drift regions are very similar, although the transition region seems to be more intricate as far as its geometry is concerned. The general form of the transition region model needs evaluation of an integral. The expression can be reduced to an algebraic one if the voltages applied to the boundaries do not differ much. This insight may enhance the evaluation speed considerably.

**Keywords**

Transistor, Integrated circuits, High-voltage LDMOS device, Compact Model, Transition Region, Drift Region, Thin-layer Approximation, Depletion Layer

## 1 Introduction

An integrated circuit (IC) consists of many thousands of semiconductor devices (transistors). In practice, there is an urgent need for mathematical models of transistors, since such models allow to simulate the behavior of an IC. The physics underlying a semiconductor is reasonably well understood, so finite-element methods may be formulated that in principle may be used for simulation. Finite-element methods, however, require a lot of computing time and memory, and for a full IC with its many transistors a finite-element model per separate transistor is therefore not manageable. Instead we would like to have a compact model for a transistor with the following properties:

- the model provides a simple-to-evaluate relation between the voltages and currents at designated places in the transistor;
- the model is scalable, that is, its physical parameters and geometry may be varied such that a large class of transistors is described.

In the following section we describe the LDMOS (Lateral Double-Diffused Metal Oxide Silicon) device, used for high voltages, in some detail and specify the parameters involved. The overall model of the device will be a combination of models for various regions in the device. We identify two such regions, the *drift* region and the *transition* region. In Section 3 we review a one-dimensional depletion layer model. This is a building block for a model for the drift region, which we consider in Section 4. Then, in Section 5 we take a closer look at the transition region, which is the region of main concern. In the process of modelling several simplifying assumptions are made along the way. The present work is not concerned with the investigation of the quality of the model developed for the transition region.

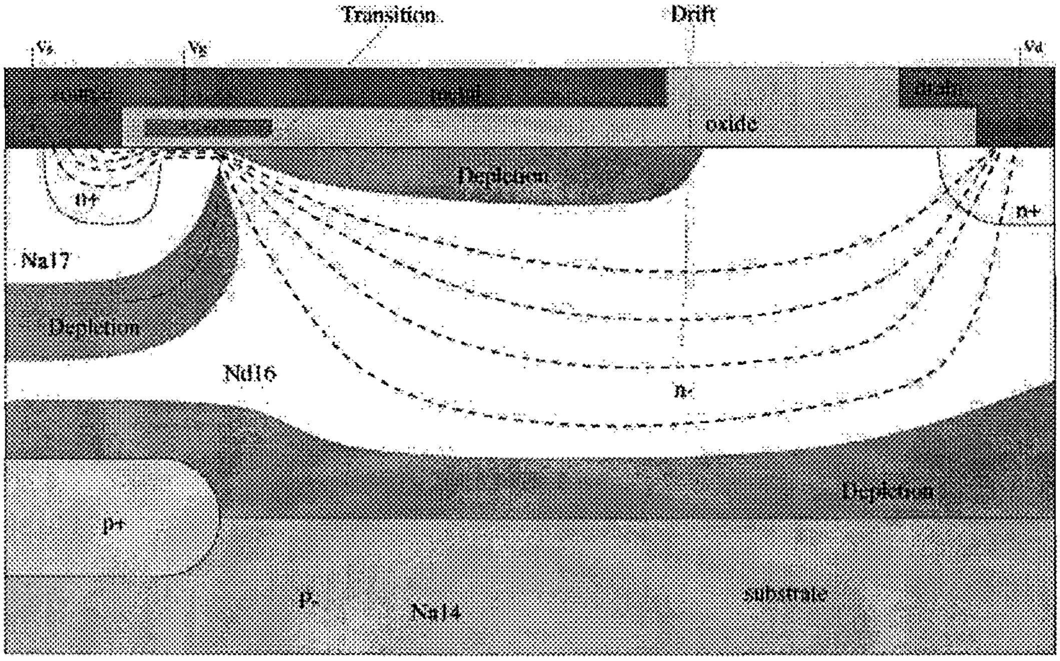

Figure 1: An LDMOS device.

## 2 An LDMOS device

Figure 1 shows a cross–section of an LDMOS device. In the top part a strip of oxide separates two strips of metal called the *source* (on the left) and the *drain* (on the right). If we set a voltage $V_d - V_s > 0$ across the two metal strips electrons will move from source to drain, and hence, an electric current will flow from right to left. The current lines are depicted as dashed curves in Figure 1. The current flows in the white regions which contain silicon. In fact, all material below the strip of oxide is inhomogeneous silicon, where the inhomogeneity is due to a variable amount of doping. In Figure 1 the concentrations of doping are denoted by $n^-$ and $n^+$ for n–doped material ($n$–material), and by $p^+$ and $p^-$ for p–doped material ($p$–material). We come back to this point in more detail in Section 3. For the moment it is enough to know that a so–called *depletion layer* is formed at places where differently doped materials meet. In Figure 1 these are the dark grey layers. They act as barriers through which only a negligible amount of current can flow. There is also a depletion layer just below the strip of oxide. The size of the depletion layers depends on the voltages, so by changing the various voltages it is possible to shrink or enlarge the depletion layers, thereby modifying the shape of the channel through which current can flow.

In the device we identify a *drift* region (a large region in the center of the device) and a tiny *transition* region (below the gate, see Figure 1). Their respective geometries differ a lot and as a result the models for them differ as well.

For the physical background of the system under consideration we refer to references [1]–[4]. Throughout we make use of the so–called drift-diffusion model, which involves the following equations:

$$\begin{aligned}
\mathbf{E} &= -\nabla\psi, \\
\nabla\cdot\mathbf{J} &= 0, \\
\mathbf{J} &= \underbrace{kT\mu_n\nabla n}_{\text{diffusion term}} + \underbrace{q\mu_n n\mathbf{E}}_{\text{drift term}}, \\
-\nabla(\epsilon\nabla\psi) &= \rho \;= q(p-n+D), \qquad (D = N_d - N_a)\ .
\end{aligned}$$

Here, $\mathbf{E}$ is the electric field intensity, $\psi$ is the potential, $\mathbf{J}$ is the current density, $k$ is Boltzmann's constant, $T$ is the temperature (in Kelvin), $\mu_n$ is the mobility of the electrons, $n$ is the free–electron concentration, $q$ is the electron charge, and $\epsilon$ is the permittivity of the material. The charge density, denoted by $\rho$, depends linearly on $p$, $n$, the concentrations of holes and free electrons, and on the doping concentration $D$ (for a $p$–material $N_d = 0$, so $D = -N_a$, whereas for an $n$–material $N_a = 0$, and $D = N_d$. In the model it is assumed that no recombination occurs; this is expressed by the equation $\nabla\cdot\mathbf{J} = \mathbf{0}$. Some typical values and ranges of the physical quantities are listed in Table 1.

| | | |
|---|---|---|
| $T \approx 300$ K | $k = 1.38\cdot10^{-23}$ J/K | $q = 1.602\cdot10^{-19}$ C |
| $N_a = 10^{14}$ cm$^{-3}$ | $n_i = 1.45\cdot10^{10}$ cm$^{-3}$ (Silicium) | $N_d = 10^{16}$ cm$^{-3}$ |
| $\mu_n = 1190$ cm$^2$/(Vs) | $\epsilon_{\text{ox}} = 0.345\cdot10^{-12}$ C/(Vcm) | $\epsilon_{\text{si}} = 1.036\cdot10^{-12}$ C/(Vcm) |
| $V_d - V_s = 12$ or 60V | $V_g - V_s = 12$ or 60V | |

Table 1: Typical values and ranges of the physical quantities.

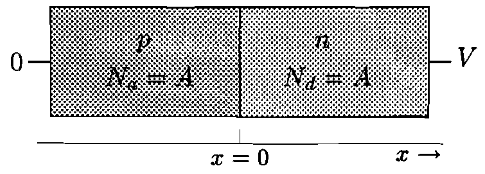

Figure 2: $p$-material meets $n$-material.

# 3 Depletion layers in doped material – one-dimensional case

In this section we review what happens if differently doped materials are brought in contact with each other. For simplicity we consider here the one-dimensional case of a $p$–material in the region $x < 0$ and an $n$–material in the region $x > 0$; see Figure 2. The two materials have opposite constant doping concentrations $\pm A$, so that $D = A\,\text{sgn}(x)$. We assume that a voltage difference $V$ is applied over the two materials joined together. At the right end ($x \rightarrow \infty$) the voltage is $V$, at the left end ($x \rightarrow -\infty$) the voltage is 0. Upon contact, electrons move until after a short while a steady state is reached at which $\mathbf{J} = \mathbf{0}$. From the drift-diffusion model we then infer that

$$kT\mu_n\nabla n + q\mu_n n(-\nabla\psi) = \mathbf{0}. \tag{1}$$

In our one-dimensional configuration where $\psi(x)$ and $n(x)$ only depend on $x$ it follows that $n(x) = c\exp\left[\frac{q}{kT}\psi(x)\right]$ for some constant $c$. In the steady state there is still a small percentage of the electrons that moves freely. The concentration of these electrons is denoted by $n_i$ and for silicon $n_i = 1.45 \cdot 10^{10}$ cm$^{-3}$. This we can exploit for the determination of $c$. For large positive $x$ the voltage $\psi(x)$ is nearly constant and close to $V$, hence, in the $n$-material we have $n(x) = n_i \exp\left[\frac{q}{kT}(\psi(x) - V)\right]$. Similarly, for large negative $x$ the voltage $\psi(x)$ is nearly constant and close to 0. Hence in the $p$-material, where $q \rightarrow \; -q$, we have $p(x) = n_i \exp\left[-\frac{q}{kT}\psi(x)\right]$. Inserting this into the drift-diffusion model we are led to the equation

$$-\epsilon\psi''(x) = \rho(x) = qn_i\left\{\exp\left[-\frac{q}{kT}\psi(x)\right] - \exp\left[\frac{q}{kT}(\psi(x) - V)\right] + \frac{A}{n_i}\operatorname{sgn}(x)\right\}. \qquad (2)$$

As $x \rightarrow \pm\infty$ one has $\rho \rightarrow 0$. On neglecting exponentially small terms it follows that

$$\psi(-\infty) = -\frac{kT}{q}\log\left(\frac{A}{n_i}\right), \qquad \psi(+\infty) = V + \frac{kT}{q}\log\left(\frac{A}{n_i}\right). \qquad (3)$$

Next we multiply both sides of (2) by $\psi'(x)$ and integrate with respect to $x$. As a result we find

$$\begin{aligned}-\frac{1}{2}\epsilon(\psi'(x))^2 \;&=\; -kTn_i\exp\left[-\frac{q}{kT}\psi(x)\right] - kTn_i\exp\left[\frac{q}{kT}(\psi(x) - V)\right] \\ &\quad +qA\psi(x)\operatorname{sgn}(x) + C_\pm, \qquad (4)\end{aligned}$$

with integration constants $C_-$ for $x < 0$, and $C_+$ for $x > 0$. These integration constants are determined by evaluating (2) at $x = \pm\infty$, where $\psi'(\pm\infty) = 0$. By use of the values of $\psi(\pm\infty)$ found above, we obtain

$$C_- = kTA\left(1 - \log\left(\frac{A}{n_i}\right)\right), \qquad C_+ = kTA\left(1 - \log\left(\frac{A}{n_i}\right)\right) - qAV. \qquad (5)$$

For reasons of symmetry we expect that $\psi(0) = V/2$. This value can be found from the property that $\psi'(x)$ is continuous at $x = 0$. Indeed, continuity of the right-hand side of (4) at $x = 0$ implies

$$-qA\psi(0) + C_- = +qA\psi(0) + C_+ \;,$$

so that $\psi(0) = (C_- - C_+)/(2qA) = V/2$. Figure 3 shows plots of the voltage $\psi(x)$, the electric field $E(x)$, and the charge density $\rho(x)$, as functions of $x$, for $V = 0$ and $V = 10$ and a doping concentration $A = 10^{16}$ cm$^{-3}$. These plots are based on a numerical solution of (4). Note the fairly abrupt transitions from a vanishing value to non-vanishing values of the charge density $\rho$. The *depletion layer* is now defined as the interval $[-l, l]$ outside of which $\rho(x)$ is effectively zero. The value of $l$ may be determined by approximating $\rho(x)$ by a piecewise constant function of the form shown on the left of Figure 4. From (2) it follows that $\rho(0^-) = -qA$, $\rho(0^+) = +qA$. Corresponding approximations for $\rho$ and $E$ by piecewise linear and quadratic functions are obtained by integration, viz

$$E = \int_{-\infty}^{x} \frac{\rho(s)}{\epsilon}ds\;, \qquad \psi(x) = \frac{V}{2} - \int_0^x E(s)ds.$$

Plots of these approximations are shown in Figure 4. The potential $\psi(x)$ varies from $\psi(-\infty) = V/2 - qAl^2/(2\epsilon)$ till $\psi(+\infty) = V/2 + qAl^2/(2\epsilon)$. Hence by comparison with the values of $\psi(\pm\infty)$ found before, we have

$$\frac{qA}{\epsilon}l^2 = \psi(+\infty) - \psi(-\infty) = V + 2\frac{kT}{q}\log\left(\frac{A}{n_i}\right). \qquad (6)$$

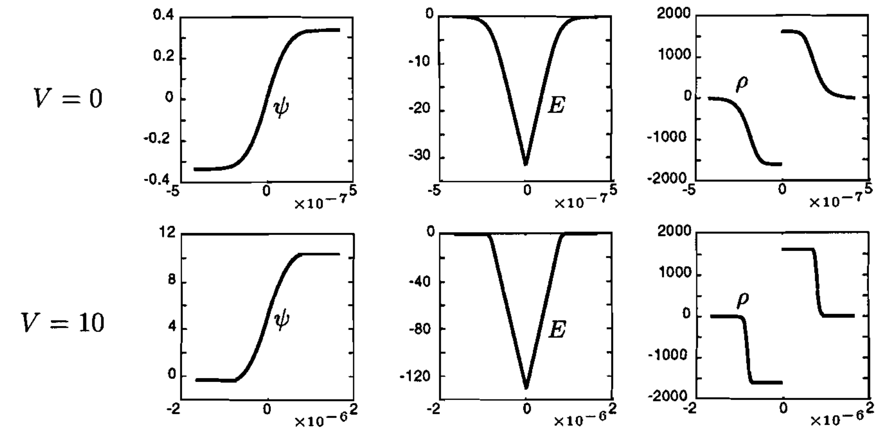

Figure 3: Plots of $\psi$, $E$ and $\rho$ across a depletion layer with $V = 0$(top) and $V = 10$ (bottom).

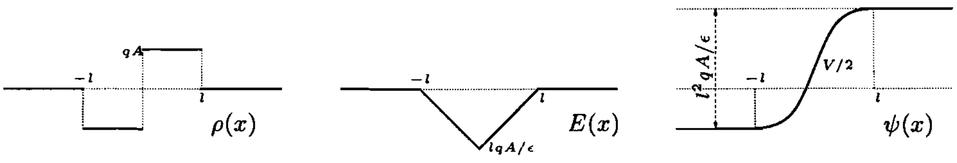

Figure 4: Figure 4: Plots of approximations of $\rho(x)$, $E(x)$, and $\psi(x)$ across a depletion layer.

This relation leads to the following expression for the width of the depletion layer as a function of the applied voltage:

$$l = \sqrt{\frac{\epsilon}{qA}(V + \psi_0)}, \qquad \psi_0 := 2\frac{kT}{q}\log\left(\frac{A}{n_i}\right). \tag{7}$$

So far we considered the case of symmetric doping: $N_d = N_a = A$. For general $N_d$ and $N_a$ the depletion layer is not symmetric although located around $x = 0$. It can be shown that the widths of the layers to the left and to the right of $x = 0$ are given by

$$l_a = \sqrt{\frac{\epsilon}{qN_a}\frac{2N_d}{N_a + N_d}(V + \psi_0)}, \qquad l_d = \sqrt{\frac{\epsilon}{qN_d}\frac{2N_a}{N_a + N_d}(V + \psi_0)}\ , \tag{8}$$

in which

$$\psi_0 = \frac{kT}{q}\left(\log\left(\frac{N_d}{n_i}\right) + \log\left(\frac{N_a}{n_i}\right)\right). \tag{9}$$

Finally, we consider a depletion layer from $x = 0$ till $x = l_s$, consisting of an $n-$material only. In this case, it can be shown that

$$l_s = \sqrt{\frac{2\epsilon}{qN_d}(V - V_0)}\ , \tag{10}$$

where $V_0$ and $V$ are the voltages at $x = 0$ and $x \to \infty$, respectively. For the derivation of this expression we refer to the analogous derivation of the expression (32) of $\vartheta_s$ in Section 5 ((28)-(31)).

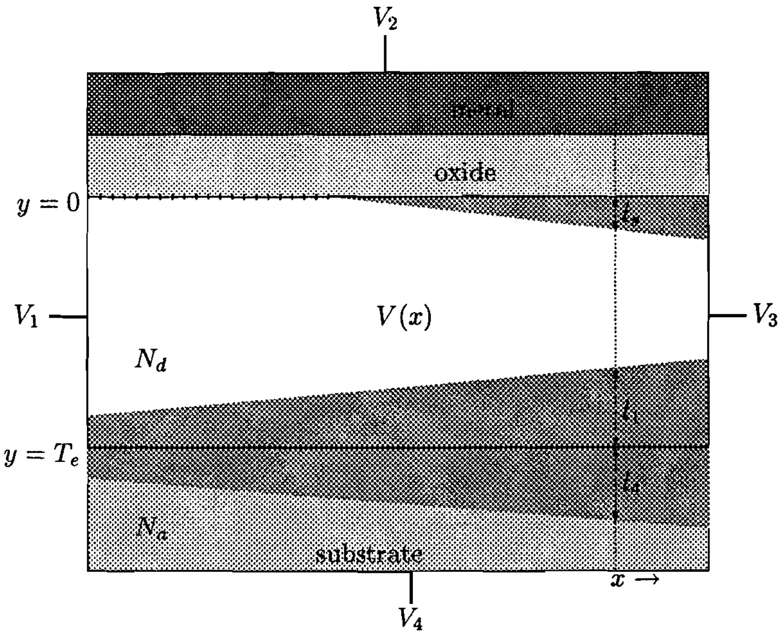

Figure 5: The drift region.

# 4 The drift region

With the depletion model into effect we analyze the drift region. This is the $n^-$-doped region $G_d$ indicated in Figure 1 and described by $G_d := \{x, y | \, x_1 < x < x_3 \, , \, l_s(x) < y < T_e - l_1(x)\}$. In $G_d$ the current roughly flows in the horizontal direction, from right to left, which is taken as the negative $x$-direction. The drift region is shown schematically in Figure 5; the four constant voltages $V_i$ along the boundaries are assumed to be known. The aim is to relate the total current $I$ (assumed to be constant), flowing from left to right through the drift region, to the voltages $V_i$. As the name suggests, in this region the effect of drift is assumed to outweigh the effect of diffusion. For the calculation of $I$ we need the voltage $V(x,y)$ in the whole region depicted in Figure 5, so not only in $G_d$, but also in the metal and oxide layers and in the three depletion layers (of widths $l_s$, $l_1$ and $l_4$). In these different regions different asymptotics apply.

In the layers, a thin-layer-approximation may be applied, implying that the Laplace operator $\triangle V(x,y)$ in the layers reduces to

$$\triangle V \approx \frac{\partial^2 V(x,y)}{\partial y^2} \, .$$

This can be seen as follows. Let $l$ be a characteristic length parameter for the width of the depletion layer and assume $l \ll L = x_3 - x_1$. Scale the coordinates $x$ and $y$ such that $x = L\hat{x}$, $y = l\hat{y}$. Then the Laplace operator for $V$ can be written in terms of $\hat{x}$ and $\hat{y}$ as

$$\triangle V = \frac{1}{L^2}\frac{\partial^2 V(x,y)}{\partial \hat{x}^2} + \frac{1}{l^2}\frac{\partial^2 V(x,y)}{\partial \hat{y}^2} = \frac{1}{l^2}\left[\left(\frac{l}{L}\right)^2 \frac{\partial^2 V(x,y)}{\partial \hat{x}^2} + \frac{\partial^2 V(x,y)}{\partial \hat{y}^2}\right]$$

$$= \frac{1}{l^2}\frac{\partial^2 V(x,y)}{\partial \hat{y}^2}\left(1 + O((l/L)^2)\right) \, ,$$

which explains the approximation used above.
The consequence is that, for fixed $x$, we may now consider $V$ as a function of $y$ only, and that we may use here the results of the one-dimensional models derived in Section 3.

At both sides of the interface $y = T_e$ between the $n^-$-doped drift region and the $p^-$-doped substrate a depletion layer will occur, the width of which is dependent on the voltage drop over the interface. Now suppose $V_3 > V_1 > V_4$. Then the width of the depletion layer near the right boundary $x = x_3$ (the boundary with potential $V_3$ in Figure 5) is larger than the width of the depletion layer near the left boundary $x = x_1$ (the boundary with potential $V_1$ in Figure 5). This will make the channel for the current narrowing towards the right. Under the oxide a similar depletion layer is formed, which also contributes to the reduction of the channel width towards the right. Consequently, the Ohmic resistance along the channel depends on the coordinate $x$.

We see that the widths of the depletion layers change, but we now assume that they change 'slowly', in so far that $|l'(x)| \ll 1$, for all $x \in (x_1,\ x_3)$. For a straight channel, the voltage in the channel will be independent of $y$ (i.e. $V = V(x)$, then). Hence, if we assume that $|l'(x)| = O(\delta),\ 0 < \delta \ll 1$, then it is also reasonable to assume that

$$V(x,y) = V(x)(1 + O(\delta)), \qquad \text{as } (x,y) \in G_d\ .$$

Thus, for small $\delta$, we have for the voltage in the drift region $G_d$:

$$V(x,y) = V(x) + \delta V_r(x,y) \ \rightarrow V(x), \qquad \text{for } \delta\ \rightarrow\ 0\ . \tag{11}$$

The widths of the two depletion layers in the drift region, denoted by $l_s$ and $l_1$, are dependent on $x$; see Figure 5. In fact, the widths depend on $x$ only via the voltage $V$ in the channel, which in its turn depends on $x$. Indeed, in the previous section we showed that (see (8))

$$l_1(x) = \sqrt{\frac{\epsilon_{\text{si}}}{qN_d}\frac{2N_a}{N_a + N_d}(V(x) - V_4 + \psi_0)} = \hat{l}_1(V(x)), \tag{12}$$

where

$$\psi_0 = \frac{kT}{q}\left(\log\left(\frac{N_d}{n_i}\right) + \log\left(\frac{N_a}{n_i}\right)\right)\ . \tag{13}$$

In a more or less analogous way, we can calculate $l_s(x)$. For this we start from (10). In this expression, $V_0$ is the voltage at $x = 0$, which is built up from the prescribed voltage $V_2$ and the potential jump $\psi_{\text{ox}}$ over the oxide layer, so

$$V_0 = V_2 + \psi_{\text{ox}}\ .$$

However, $\psi_{\text{ox}}$ also depends on $l_s$, as can be shown as follows.
On the interface between the metal layer (conductor) and the oxide layer (dielectric) there will be a surface charge density, say $Q_2$. Moreover, let the total charge per unit of length in the $x$-direction in the $l_s$-depletion layer be $Q_s$, so $Q_s = qN_dl_s$. Then, due to the global charge neutrality, we have

$$Q_2 = -Q_s = -qN_dl_s\ . \tag{14}$$

The potential jump over the oxide layer, which has width $T_{\text{ox}}$ and permittivity $\epsilon_{\text{ox}}$, is

$$\psi_{\text{ox}} = \frac{Q_2}{\epsilon_{\text{ox}}}\,T_{\text{ox}} = \frac{qN_d}{C_{\text{ox}}}\ l_s\ , \tag{15}$$

where

$$C_{\text{ox}} = \frac{\epsilon_{\text{ox}}}{T_{\text{ox}}} \; . \tag{16}$$

Hence,

$$V_0 = V_2 + \frac{qN_d}{C_{\text{ox}}}\, l_s \; . \tag{17}$$

Substituting this result into (10)(with $\epsilon = \epsilon_{\text{si}}$), we obtain

$$l_s^2 = \frac{2\epsilon_{\text{si}}}{qN_d}\left(V - V_2 - \frac{qN_d}{C_{\text{ox}}}\, l_s\right) \; , \tag{18}$$

resulting in the following expression for $l_s$:

$$l_s(x) = \sqrt{\frac{2\epsilon_{\text{si}}}{qN_d}}\left(\sqrt{V - V_2 + V_{\text{si}}} - \sqrt{V_{\text{si}}}\right) = \hat{l}_s(V) \; , \tag{19}$$

where

$$V_{\text{si}} = \frac{q\epsilon_{\text{si}}N_d}{2C_{\text{ox}}^2} \; . \tag{20}$$

Let $J(x)$ be the current density in the channel in the $x-$direction. By use of the drift-diffusion model with the diffusion term neglected, we find that the channel current density at $x$ is given by

$$J(x) = -q\mu_n N_d \frac{dV}{dx}(x) \; . \tag{21}$$

The total current $I(x)$ flowing through a cross-section of the channel at $x$ is then given by

$$\begin{aligned} I(x) &= W \int_{l_s(x)}^{T_e - l_1(x)} J(x)\, dy \\ &= -Wq\mu_n N_d [T_e - l_1(x) - l_s(x)] \frac{dV}{dx}(x) \; , \end{aligned} \tag{22}$$

where $W$ is the width of the channel in the direction perpendicular to the $x-y-$plane. The relation between $I$ and $V$ still depends on $x$. However, the channel current $I$ is independent of $x$, since there is no accumulation of charge. Therefore we have the obvious relation

$$I = \frac{1}{x_3 - x_1}\int_{x_1}^{x_3} I(x)dx = -\frac{W\mu_n}{x_3 - x_1}\int_{V_1}^{V_3} qN_d(T_e - \hat{l}_1(V) - \hat{l}_s(V))\, dV. \tag{23}$$

The integrand is a known function of $V$, hence, the total current $I$ may be calculated as a function of $V_1$ and $V_3$: $I = I(V_1, V_3)$. To get some insight into the function $I(V_1, V_3)$, we expand it about the equilibrium point where all boundary voltages are the same: $V_1 = V_2 = V_3 = V_4$. To that end we write

$$qN_d(T_e - \hat{l}_1(V) - \hat{l}_s(V)) = \underbrace{qN_d\big(Te - \hat{l}_1(V_4)\big)}_{q_i} - \underbrace{qN_d\big(\hat{l}_1(V) - \hat{l}_1(V_4)\big)}_{q_b(V)} - \underbrace{qN_d\hat{l}_s(V)}_{q_s(V)} . \tag{24}$$

By construction, both $q_s = 0$ and $q_b = 0$ if $V_1 = V_2 = V_3 = V_4$. Therefore around the equilibrium point we have

$$\begin{aligned} I &= -\frac{W\mu_n}{x_3 - x_1} \int_{V_1}^{V_3} (q_i - q_b(V) - q_s(V))\, dV \\ &= -\frac{V_3 - V_1}{R_{\text{on}}} + \text{higher-order terms}, \end{aligned} \tag{25}$$

in which

$$R_{\text{on}} := \frac{x_3 - x_1}{W\mu_n q_i} \tag{26}$$

may be interpreted as the ohmic resistance near equilibrium.

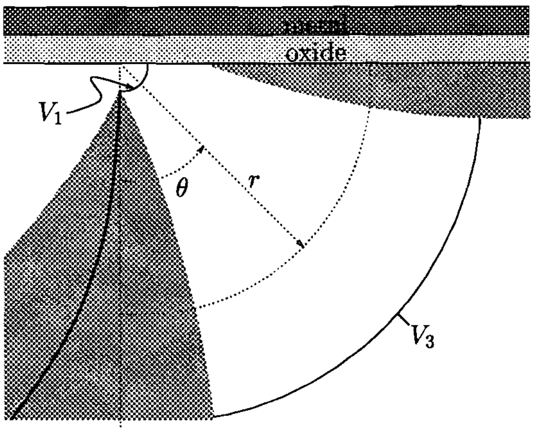

Figure 6: The transition region

# 5 The transition region

The transition region is a small wedge-shaped region of about 2$\mu$m x 2$\mu$m below the gate, as shown in Figure 1. In the transition region we use the polar coordinates $r, \theta$; see Figure 6. The true transition region $G_t$ (i.e. the white region in Figure 1) is surrounded by two depletion layers, both of $n-$type. The first layer is connected to the oxide layer and runs from $\theta = 0$ to $\theta = \vartheta_s(r)$; thus the thickness of the layer at radius $r$ is $r\vartheta_s(r)$, which should be compared to $l_s(x)$ for the drift region. The second layer runs from $\theta = \pi/2 - \vartheta_1(r)$ to $\theta = \pi/2$; here the width $r\vartheta_1(r)$ is comparable to $l_1(x)$. To the left of this layer there is a third layer, but now of $p-$type. This layer runs from $\theta = \pi/2$ to $\theta = \pi/2 + \vartheta_4(r)$ with width $r\vartheta_4(r)$ comparable to $l_4(x)$. As before, in these depletion layers $\mathbf{J} = \mathbf{0}$ and the charge density is $qN_d$ in the $\vartheta_s-$ and $\vartheta_1-$layers and $qN_a$ in the $\vartheta_4-$layer.
The true transition region is given by

$$G_t = \{r, \theta |\; r_1 < r < r_3,\; \vartheta_s(r) < \theta < \pi/2 - \vartheta_1(r)\}\ .$$

In this region a current $\mathbf{J}$ runs mainly in radial direction and dependent on $r$, while the charge density $\rho$ vanishes, implying that $\triangle V = 0$. Here, $V = V(r, \theta)$ is the voltage in $G_t$, which has prescribed values $V_1$ and $V_3$ at the boundaries $r = r_1$ and $r = r_3$, respectively; see Figure 6.

The analysis of this wedge-shaped transition region is in main lines analogous to that of the rectangular drift region. Firstly, in the depletion layers we may apply a thin-layer-approximation, stating that we may consider the voltage at fixed $r$ as a function of $\theta$ only, resulting in the following equation for $V$:

$$\frac{1}{r^2}\frac{\partial^2 V(r,\theta)}{\partial\theta^2} = -\frac{\rho}{\epsilon_{\text{si}}} \ . \tag{27}$$

In the first depletion layer this equation becomes

$$\frac{\partial^2 V_s(r,\theta)}{\partial\theta^2} = -\frac{qN_d}{\epsilon_{\text{si}}}\, r^2 \ , \qquad 0 < \theta < \vartheta_s(r) \ , \tag{28}$$

with the boundary conditions ($V_2$ is again the voltage applied to the oxide layer)

$$V_s(r,0) = V_2 + \psi_{\text{ox}} \ , \qquad V_s(r,\vartheta_s) = V \ , \tag{29}$$

where

$$\psi_{\text{ox}} = \psi_{\text{ox}}(r) = \frac{qN_d}{C_{\text{ox}}}\, r\vartheta_s(r) \ , \tag{30}$$

and $V(= V(r))$ is the voltage in $G_t$.
The solution of this problem reads

$$V_s(r,\theta) = \frac{qN_d}{2\epsilon_{\text{si}}}\, r^2(\vartheta_s\theta - \theta^2) + V\,\frac{\theta}{\vartheta_s} + (V_2 + \psi_{\text{ox}})\left(1 - \frac{\theta}{\vartheta_s}\right) \ . \tag{31}$$

The angular width $\vartheta_s$ follows from the requirement

$$\frac{\partial V_s}{\partial\theta}(r,\vartheta_s) = 0 \ ,$$

yielding

$$\vartheta_s(r) = \frac{1}{r}\sqrt{\frac{2\epsilon_{\text{si}}}{qN_d}}\left(\sqrt{V - V_2 + V_{\text{si}}} - \sqrt{V_{\text{si}}}\right) = \frac{1}{r}\,\hat{l}_s(V) \ , \tag{32}$$

with $V_{\text{si}}$ as in (20).
It is now obvious that we can find $\vartheta_1(r)$ in accordance with (12) and (13) as

$$\vartheta_1(r) = \frac{1}{r}\sqrt{\frac{\epsilon_{\text{si}}}{qN_d}\frac{2N_a}{N_a + N_d}(V(r) - V_4 + \psi_0)} = \frac{1}{r}\,\hat{l}_1(V), \tag{33}$$

Secondly, in the transition region $G_t$ we assume that the voltage is independent of $\theta$, so that $V = V(r)$. This assumption can be justified as follows. Since $\rho = 0$ in $G_t$, $V$ satisfies $\triangle V = 0$, and then it follows from a separation–of–variables argument that $V$ is of the form

$$V(r,\theta) = a_0 + a_1\log(r) + \sum_{n\in\mathbb{Z}} r^n(c_n\cos(n\theta) + d_n\sin(n\theta)). \tag{34}$$

If the boundaries of the depletion layers would be perfectly straight (i.e. $\vartheta_s(r) = \vartheta_s$ and $\vartheta_1(r) = \vartheta_1$), the boundary conditions for $V$ would read: $V(r_1,\theta) = V_1$; $V(r_3,\theta) = V_3$; $\partial V/\partial\theta = 0$ for $\theta = \vartheta_s$ and $\theta = \vartheta_1$. This would imply that $c_n = d_n = 0$, for all $n$, so $V = V(r)$. Hence, when we assume that the boundaries of the depletion layers are 'slowly varying', it is allowed (in accordance with the approach for the drift region; compare to (11)) to take

$$V(r,\theta) = V(r) + \delta V_r(r,\theta) \ .$$

Thus, with $V(r_1) = V_1$ and $V(r_3) = V_3$ we obtain (up to $O(\delta)$)

$$V(r) = V_1 + \frac{V_3 - V_1}{\log(r_3/r_1)} \log\left(\frac{r}{r_1}\right) . \tag{35}$$

Similar to (21) and (22), the current density and the total current through the channel in the transition region are now given by ($\mathbf{J} = J(r)\mathbf{e}_r$)

$$J(r) = -q\mu_n N_d \, \frac{dV(r)}{dr} \ , \tag{36}$$

and

$$\begin{aligned} I(r) &= W \int_{\vartheta_s(r)}^{\frac{\pi}{2}-\vartheta_1(r)} J(r)\, r d\theta \\ &= -Wq\mu_n N_d \left[\frac{\pi}{2} - \vartheta_1(r) - \vartheta_s(r)\right] r\frac{dV(r)}{dr} \\ &= -Wq\mu_n N_d \left[\frac{\pi}{2} r\frac{dV(r)}{dr} - \hat{l}_1(V_4)\frac{dV(r)}{dr}\right] \\ &\quad +Wq\mu_n N_d \left[\hat{l}_1(V) - \hat{l}_1(V_4)\right] \frac{dV(r)}{dr} \\ &\quad +Wq\mu_n N_d \left[\hat{l}_s(V)\frac{dV(r)}{dr}\right] \\ &= -W\mu_n \left[q_i(r) - (q_b(V) + q_s(V))\frac{dV}{dr}\right] \ , \end{aligned} \tag{37}$$

where $q_b$ and $q_s$ are defined in (24), and $q_i$ is given by

$$q_i(r) = qN_d \left[\frac{\pi}{2} r - \hat{l}_1(V_4)\right] \frac{dV(r)}{dr} \ , \tag{38}$$

while it follows from (33) that

$$\hat{l}_1(V_4) = \sqrt{\frac{\epsilon_{\mathrm{si}}}{qN_d} \frac{2N_a}{N_a + N_d} \psi_0} \ . \tag{39}$$

Since there is no accumulation of charge we again argue that

$$\begin{aligned} I &= \frac{1}{r_3 - r_1} \int_{r_1}^{r_3} I(r)\, dr \\ &= -\frac{W\mu_n}{r_3 - r_1} \frac{qN_d(V_3 - V_1)}{\log(r_3/r_1)} \int_{r_1}^{r_3} \left(\frac{\pi}{2} - \frac{\hat{l}_1(V_4)}{r}\right) dr \\ &\quad -\frac{W\mu_n}{r_3 - r_1} \int_{V_1}^{V_3} (q_b(V) + q_s(V))\, dV \ . \end{aligned} \tag{40}$$

Around the equilibrium point $V_1 = V_2 = V_3 = V_4$, where $q_b = q_s = 0$, the total current is given by

$$\begin{aligned} I &= -\frac{\pi}{2} \frac{\mu_n q N_d W}{\log(r_3/r_1)} \left[1 - \frac{2}{\pi}\hat{l}_1(V_4)\frac{\log(r_3/r_1)}{r_3 - r_1}\right] (V_3 - V_1) + \text{higher–order terms} \\ &= -\frac{V_3 - V_1}{\hat{R}_{\mathrm{on}}} + \text{higher–order terms} \ , \end{aligned} \tag{41}$$

in which the ohmic resistance is given by

$$\frac{1}{\hat{R}_o n} = \frac{\pi}{2} \frac{\mu_n q N_d W}{\log(r_3/r_1)} \left[1 - \frac{2}{\pi}\hat{l}_1(V_4)\frac{\log(r_3/r_1)}{r_3 - r_1}\right]. \tag{42}$$

With this final result, the total current in the transition region is written in a form similar to the one obtained in the drift region.

According to the numerical values of Table 1, and with $r_3 = 2\mu$m, the second term between the square brackets on the right-hand side of (41) is of the order of $10^{-1}$. Hence, it is less than 1, but not really negligible with respect to 1.

# 6 Concluding remarks

A high–voltage MOS(metal–oxide–silicon) device consists of several regions, such as the body region, the drift region, and the transition region. A compact model for the device as a whole requires coupling of the compact models for these separate regions. Up to now, one uses for the transition region a model that has been developed for the drift region and is adapted to the transition region via ad-hoc considerations. The main goal of the present project is the derivation of a reliable compact model for the transition region from first principles. Since the drift and transition regions have the same characteristics and differ mainly in geometry, it is to be expected that many similarities exist between the corresponding models. That is why the Study-group started with rederiving the drift region model in full detail, in order to find which mechanisms are most important. For this, the basic principles described in [1]–[4] formed the starting points.

An important ingredient of a compact model for the drift region is the thin-layer-approximation for the depletion layers. Another assumption is that the widths of these layers vary rather slowly in the longitudinal direction of the channel. This implies that, to lowest order, the voltage in the drift region is a function of the horizontal position only. This approach resulted in the compact model embodied in expression (23), which relates the current through the channel to the voltages applied at the boundaries. This model contains an integral. If the boundary voltages are nearly equal, the model is described by expression (25), which is algebraic and extremely simple to evaluate.

The derivation of the drift region model provided the insights for the derivation of the transition region model. Here, polar coordinates ($r$, $\theta$) are used. It is shown that the voltage in the transition region is a function of $r$ only, if the form of the depletion layers is wedge-like with straight boundaries, i.e. $\theta$ = constant along these boundaries. It is assumed that the deviations from straight lines are small, so that, to lowest order, the $\theta$–dependence of the voltage may be ignored. This allows for a derivation along the same lines as followed for the drift region. The resulting compact model given in (40), relates the current through the transition region to the voltages applied to the boundaries of this region. If the latter voltages are nearly equal, the model is described by (41). The latter algebraic expression, which is very easy to evaluate, shows that, to lowest order, the current is given by Ohm's law: it is proportional to the voltage difference $V_3 - V_1$ over the transition channel, and the resistance, given by (42), is a function of geometry and the other boundary voltages.

We conclude that both the drift and the transition regions of the MOS-device can be adequately represented by compact models and these models are quite similar. The models, derived above, are reliable as long as in the drift region the depletion layers vary slowly in width, whereas in the transition region the boundaries of the depletion layers do not strongly deviate from straight lines. If these conditions are not fulfilled, the present models could be extended with correction terms.

## References

1. Klaassen, F.M., *Compact models for circuit simulation*, Springer, Vienna, 1989.
2. Muller, R.S. and Kamins, T.I., *Device electronics for integrated circuits*, John Wiley & Sons, New York, 1986.
3. Sze, S.M., *Physics of semiconductor devices*, John Wiley & Sons, New York, 1981.
4. Tsividis, Y., *Operation and modeling of the MOS-transistor*, Mac Graw-Hill, Boston, 1999.

# Modelling of moisture induced warp in panels containing wood fibres

D. Chandra, H.J.J. Gramberg, T. Ivashkova, W.R. Smith, A. Suryanto, J.H.M. ten Thije Boonkkamp, T. Ulicevic and J.C.J. Verhoeven

**Abstract**

In this contribution the deformation of panels, used a.o. in furniture, is discussed in relation to the moisture content. It is shown how the variations in temperature, water and resin concentrations during the pressing process of the panels can be modelled. The panel deformation is modelled using linear elasticity theory. An explicit analytical expression for the long term behaviour of the water concentration is presented.

**Keywords**

Panel deformation, Moisture content, Linear elasticity theory.

## 1 Introduction

Trespa International BV manufactures high quality panel material. This material consists of polymerised resin reinforced with wood fibres or sulphate paper. These panels may deform due to variations in moisture content and the motivation for this study is to obtain mathematical models which help to explain this deformation.

The purpose of this report is to gain a better understanding of the behaviour of TRESPA panels. The modelling concerns the three processes of resin curing during pressing, moisture movement and panel deformation. These three processes depend on each other. The resin curing provides the initial water concentration for the moisture movement model. The moisture distribution itself is an input to the panel deformation model.

The contents will now be outlined. Section 2 describes two mathematical models for the resin curing process. The first model was discussed in a previous study [3]. The second model describes the temperature, concentration of the water and resin during pressing. In Section 3, the movement of water is modelled by a linear diffusion equation. A numerical solution is included. Section 4 derives a new mathematical model for the panel deformation based on linear elasticity. After long time periods the water concentration will be linear across the panel, an analytical solution for the deformation is presented in this case.

## 2 A thermo-chemical model for resin curing

In this section we briefly describe two models for resin curing during the pressing of TRESPA-panels; a more elaborate description of the first model is given in [3]. During the manufacturing of panels, sheets of resin impregnated paper enclosed by two layers of padded paper are pressed together. A schematic, three-layer model of a panel is given in Figure 1. A panel thus has two polsters of padded paper and a core consisting of impregnated paper. During the pressing of a panel, high temperatures are applied at the boundaries $z = -h$ and $z = h$ of the panel, which causes heating of the panel. This induces a polymerization reaction in the core. Heat is released during the polymerization, which again leads to an increase of temperature in the panel. This process continues until the chemicals in the core are depleted.

## 2.1 One species model

The curing process can be described by the temperature $T$ in the panel and the concentration $C_1$ of chemicals involved in the polymerization reaction in the core. To keep the model feasible, we make the following assumptions:

1. The materials are incompressible.
2. Resin curing is a first order reaction.
3. Diffusion of resin is negligible.
4. All variables only depend on the transverse space coordinate $z$.

The governing equations are now the following. In both polsters the heat equation holds, and in our particular case is given by [2]

$$\rho c_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}\Big(\lambda \frac{\partial T}{\partial z}\Big), \tag{1}$$

with $\rho$, $c_p$ and $\lambda$ the density, specific heat and thermal conductivity, respectively, of the polster material. These variables are assumed to be constant. In the core, heat transport is coupled with the polymerization reaction, and the governing equations read

$$\rho c_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}\Big(\lambda \frac{\partial T}{\partial z}\Big) + \Delta H k C_1, \tag{2}$$

$$\frac{\partial C_1}{\partial t} = -kC_1, \tag{3}$$

with $\Delta H$ and $k$ the enthalpy of polymerization and the reaction rate, respectively. This reaction rate is given by the Arrhenius expression

$$k = A\mathrm{e}^{-E_a/RT}, \tag{4}$$

with $A$, $E_a$ and $R$ the pre-exponential factor, activation energy and gas constant, respectively.

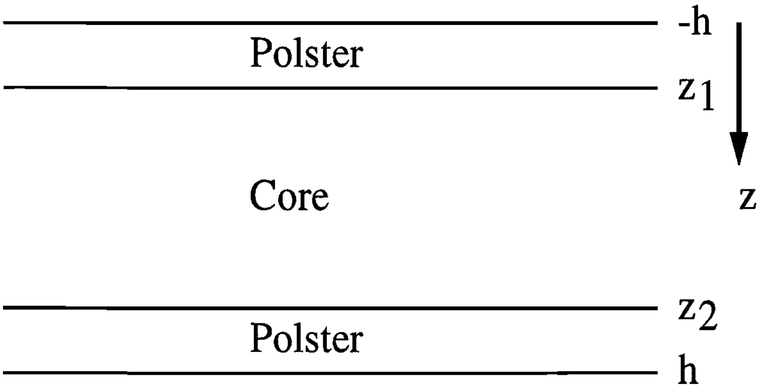

Figure 1: Schematic representation of a three-layer TRESPA-panel.

The model (1)-(4) has to be completed with initial and boundary conditions and conditions at the interfaces between core and polsters. As initial conditions, we choose a constant temperature $T$ and concentration $C_1$. At the boundaries, the temperature is given as a function of time, i.e.

$$T(-h,t) = T_1(t), \quad T(h,t) = T_2(t), \quad t > 0. \tag{5}$$

Finally, at the interfaces we impose that both the temperature $T$ as well as the heat flux $\lambda\partial T/\partial z$ are continuous. This results in the conditions

$$T(z-,t) = T(z+,t), \qquad \Big(\lambda\frac{\partial T}{\partial z}\Big)(z-,t) = \Big(\lambda\frac{\partial T}{\partial z}\Big)(z+,t), \quad z = z_1, z_2. \tag{6}$$

These conditions mean that there is no accumulation of heat at the interfaces.

As an example, we have computed a numerical solution of the system (1)-(4) using the finite difference method. More specifically, we used central differences for space discretization and the $\vartheta$-method for time integration [1]. For more details, the reader is referred to [3]. In this example, we have a constant initial temperature and concentration. Then, at $t = 0$, a high temperature is applied at the boundaries of the panel. The evolution of the temperature and concentration profiles is shown in Figure 2. Initially, the temperature increases due to conduction and heat production and at the same time the concentration decreases. When the chemical species are depleted, the temperature profile tends to the constant steady state.

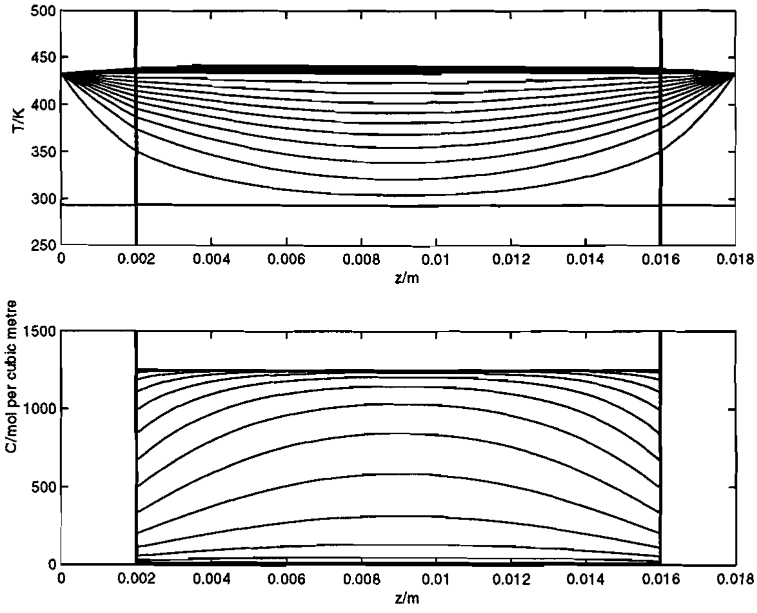

Figure 2: Temperature and concentration profiles in a TRESPA-panel.

## 2.2 Two species model

A slightly more sophisticated model for resin curing will now be introduced. The dependent variables are the temperature $T$, the concentration of water $C$ and the concentration of resin $C_1$. There are now five layers in the model as shown in Figure 3. In the polster and metal regions, the standard heat equation (1) applies. In the core, we have (2)-(3) and

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial z}\Big(D\frac{\partial C}{\partial z}\Big) + \gamma k C_1, \tag{7}$$

where $\gamma$ is a dimensionless number representing the rate at which moisture is produced relative to the rate at which chemicals involved in the polymerization are reduced. The thermal conductivity is now taken to be of the form

$$\lambda(C_1) = \begin{cases} \lambda_1 & C_1 \leq C_{crit}, \\ \lambda_2 & C_1 > C_{crit}, \end{cases}$$

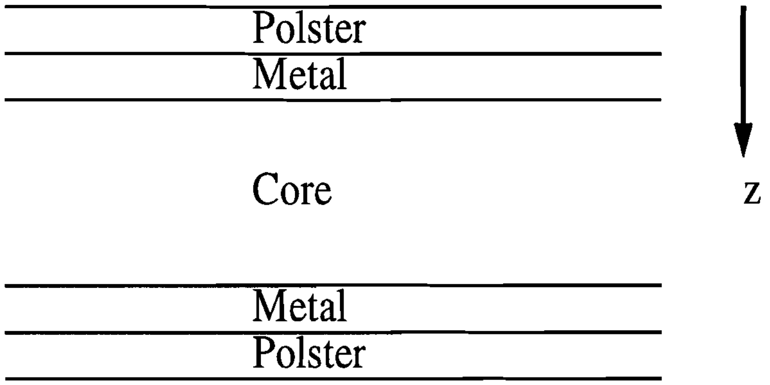

Figure 3: Schematic representation of a five-layer TRESPA-panel.

where $C_{crit}$ is an experimentally observed constant and the diffusivity of water is given by

$$D(T) = B\exp(-T_*/T)$$

where $B = 3 \times 10^{-5}\mathrm{m}^2\mathrm{s}^{-1}$ and $T_* = 5245\mathrm{K}$. The boundary conditions are (5) and the interface conditions between the core and metal are given by (6) and

$$\frac{\partial C}{\partial z}(z,t) = 0, \quad z = z_1, z_2. \tag{8}$$

# 3 Moisture transport in panels

In this section we outline a model for the transport of moisture in a panel. A non-uniform water distribution leads to non-uniform stresses in the panel, and this will lead to warping of the panel. This will be described further in the next section.

For moisture transport in panels, we adopt a particularly simple model, viz. we consider the panel as a single layer of material in which diffusion of water takes place. Further assumptions are:

1. Swelling or shrinkage in the transverse direction are negligible.
2. The material is homogeneous.
3. The concentration of water only depends on the transverse space coordinate $z$.

Under these assumptions, the diffusion of water in a panel is governed by the equation [4]

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial z}\left(D\frac{\partial C}{\partial z}\right), \tag{9}$$

with $C$ the concentration of water and $D$ the diffusion coefficient. The diffusion coefficient generally depends on the temperature $T$, but in our model we assume it to be constant. A given initial concentration $C_0(z)$ and Dirichlet boundary conditions complete the problem; we will not specify these any further.

As an example we have computed a numerical solution of (9) using central differences for space discretization and the $\vartheta$- method for time integration. The result is presented in Figure 4. This figure typically shows the evolution of concentration profiles starting from a constant initial concentration, when at $t = 0$ the boundaries are exposed to higher concentrations of water, due to moisture in the environment.

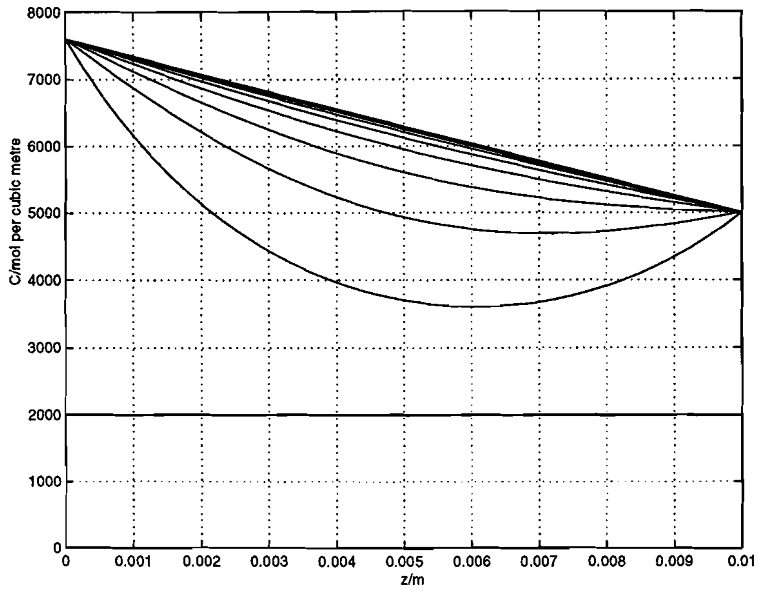

Figure 4: Water concentration profiles in a TRESPA-panel

# 4 Panel Deformation

In this section the warpage of Trespa panels due to humidity effects is studied. We assume that the Trespa panel can be modelled as a beam. This model is derived under the assumption that the material is linearly elastic [5] and that gravity is negligible.

We may assume that the centre of the beam is clamped due to symmetry considerations. The coordinate system used is sketched in Figure 5. We define the displacement vector $(u_x, u_z) = (u, w)$. There are five unknowns in the warpage problem, namely the stresses in the $x$ and $z$-direction, $t_{xx}$, $t_{xz}$ and $t_{zz}$, and the displacements $u$ and $w$. Therefore, we need five equations to determine these variables.

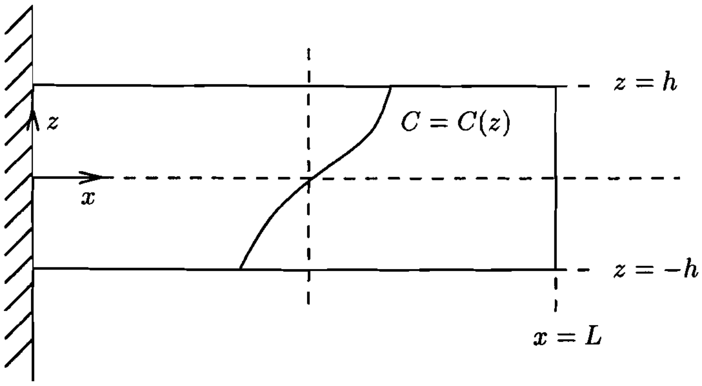

Figure 5: The geometry of the beam model.

First of all, conservation of momentum yields

$$\frac{\partial t_{xx}}{\partial x} + \frac{\partial t_{xz}}{\partial z} = 0, \tag{10a}$$

$$\frac{\partial t_{xz}}{\partial x} + \frac{\partial t_{zz}}{\partial z} = 0. \tag{10b}$$

To close the system we need three additional equations, which follow from the constitutive behaviour of the material. We define the deformation tensor as follows

$$e_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \qquad i,j = x,z. \tag{11}$$

The deformation tensor is assumed to be split up into a part which represents the deformation caused by elastic effects and a part that represents the deformation caused by expansion of the material due to swelling:

$$e_{ij} = e_{ij}^{(el)} + e_{ij}^{(sw)}. \tag{12}$$

In our model, we assume that the deformation of the panel due to humidity is linearly dependent on the concentration of moisture inside the material. Experiments carried out by Trespa International B.V. support this assumption. This leads to the the following set of equations for $e_{ij}^{(sw)}$

$$e_{xx}^{(sw)} = \alpha_x C, \tag{13a}$$

$$e_{xz}^{(sw)} = 0, \tag{13b}$$

$$e_{zz}^{(sw)} = \alpha_z C, \tag{13c}$$

where $\alpha_x$ and $\alpha_z$ represent the swelling in the $x$ and the $z$ direction, respectively. We note that this model is analogous to thermoelasticity except that in this case the strain due to moisture is anisotropic.

Because the deformations are small, we assume the material to be linearly elastic. Also we take the material to be homogeneous and isotropic with respect to elastic deformations. Therefore, we can use Hooke's law which relates the deformations to the stresses $t_{ij}$. This gives

$$e_{ij}^{(el)} = \frac{1+\nu}{E}t_{ij} - \frac{\nu}{E}\delta_{ij}t_{kk}, \tag{14}$$

where $E$ is the elasticity modulus and $\nu$ is Poisson's ratio. Substituting Eqs. (13) and (14) into Eq. (12) yields the following set of equations for the total deformations in the plate

$$e_{xx} = \frac{1+\nu}{E}t_{xx} - \frac{\nu}{E}(t_{xx} + t_{zz}) + \alpha_x C, \tag{15a}$$

$$e_{xz} = \frac{1+\nu}{E}t_{xz}, \tag{15b}$$

$$e_{zz} = \frac{1+\nu}{E}t_{zz} - \frac{\nu}{E}(t_{xx} + t_{zz}) + \alpha_z C. \tag{15c}$$

Combining Eqs. (11) and (15) leads to the following set of equations which relate the stresses $t_{ij}$ to the displacements $u$ and $w$,

$$t_{xx} = \frac{E}{1-\nu^2}\left(\frac{\partial u}{\partial x} + \nu\frac{\partial w}{\partial z} - (\alpha_x + \nu\alpha_z)C\right), \tag{16a}$$

$$t_{xz} = \frac{E}{1+\nu}\frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right), \tag{16b}$$

$$t_{zz} = \frac{E}{1-\nu^2}\left(\frac{\partial w}{\partial z} + \nu\frac{\partial u}{\partial x} - (\alpha_z + \nu\alpha_x)C\right). \tag{16c}$$

Together with (10), this yields a coupled set of five first order linear partial differential equations. To complete this set of partial differential equations, we still need to derive boundary conditions. The boundary conditions are given by the following

$$t_{zz} = t_{xz} = 0, \qquad \text{on} \quad z = \pm h, \tag{17a}$$

$$t_{xx} = t_{xz} = 0, \qquad \text{on} \quad x = L, \tag{17b}$$

$$u = \frac{\partial w}{\partial x} = 0, \qquad \text{on} \quad x = 0. \tag{17c}$$

Here, Eqs. (17a) and (17b) are stress free boundary conditions and Eq. (17c) represents the clamped end.

Since the aspect ratio is small, we try to find a solution by assuming that the normal stress component in the $z$-direction, $t_{zz}$, is negligible. With this assumption, it follows from (10), (17a) and (17b) that also $t_{xx}$ and $t_{xz}$ vanish everywhere. Using this, (16) gives us

$$\frac{\partial u}{\partial x} = \alpha_x C, \tag{18a}$$

$$\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} = 0, \tag{18b}$$

$$\frac{\partial w}{\partial z} = \alpha_z C. \tag{18c}$$

Using $C = C(z)$, we obtain from Eqs. (18a) and (18c)

$$u = \alpha_x C(z)x + g(z), \tag{19a}$$

$$w = \alpha_z \int^{z} C(\xi)d\xi + l(x). \tag{19b}$$

The boundary condition on $u$ in Eq.(17c) gives us $g(z) \equiv 0$. Substituting Eqs. (19a) and (19b) into Eq. (18b) gives us

$$\alpha_x C'(z)x + l'(x) = 0, \tag{20}$$

everywhere. This tells us that $C$ has to be linear, i.e. $C(z) = A_1 + A_2 z$. This corresponds to the moisture profile after the panel has been exposed to a different concentrations on either face for long time periods, see Section 3. Using this, we can solve Eq. (20) for $h$, yielding

$$l(x) = -\frac{1}{2}\alpha_x A_2 x^2 + D, \tag{21}$$

where $D$ is a constant representing translation. Taking $D = 0$ we end up with the following expressions for the displacements

$$u = \alpha_x(A_1 + A_2 z)x, \tag{22a}$$

$$w = \alpha_z\left(A_1 z + \frac{1}{2}A_2 z^2\right) - \frac{1}{2}\alpha_x A_2 x^2. \tag{22b}$$

Note that the second boundary condition in Eq. (17c) is satisfied. We note that the equations in this section are linear and $w$ is only specified in terms of its derivatives, therefore the solution (22a)-(22b) is unique up to a translation in $w$. The centre line $z = 0$ is approximately a circular arc with radius given by $1/\alpha_x A_2$.

For a plate with a thickness of 1cm, and a length of 2m, the results are shown in Figure 6. We have taken the following $\alpha_x C(z) = 0.002 * (h + z)\ \mathrm{m}^{-1}$ and $\alpha_z C(z) = 0.03 * (h + z)\ \mathrm{m}^{-1}$ where $h$ is half the thickness of the plate.

The solution (22a)-(22b) only allows us to deal with the concentration of water as a linear function of $z$. One possible technique for dealing with a general form for the concentration of water is asymptotics. An analysis was undertaken with the small parameter being the square of the aspect ratio. There is a boundary layer at the edge of the beam ($x = L$). The displacements were not determined at leading order in the outer expansion. Further research is required.

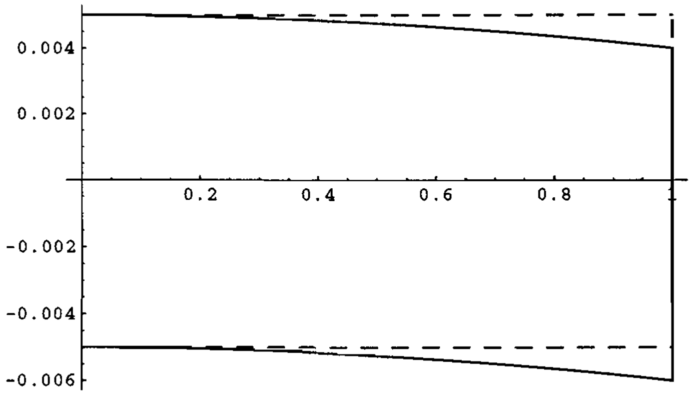

Figure 6: The warpage of a panel due to an asymmetric moisture content.

# References

[1] Morton, K.W. and Mayers, D.F., *Numerical Solution of Partial Differential Equations*, Cambridge University Press, Cambridge, 1994.

[2] Necati Özişik, M., *Heat Conduction*, John Wiley, New York, 1993.

[3] Ulicevic, T. and Ivashkova, T., *Thermo-chemical model of heat-driven polymerization in TRESPA production process*, ECMI student report 99-08, Eindhoven University of Technology, 1999.

[4] Welty, J.R., Wicks, C.E. and Wilson, R.E., *Fundametals of Momentum, Heat and Mass Transfer*, John Wiley, New York, 1984.

[5] Nadeau, G., *Introduction to Elasticity*, Holt, Rinehart and Winston, New York, 1964.

# Windtunnel model position and orientation

R. Stoffer, C. Stolk, S.W. Rienstra, J.K.M. Jansen

**Abstract**

In this contribution the determination of the position of moving and deforming objects in windtunnels from CCD camera information is studied. An analytical approach is discussed which solves the problem directly from manipulating nonlinear distance formulae. Also a least-squares approach is given, which is most convenient to implement from a numerical point of view.

**Keywords**

Windtunnel, CCD-camera, Position data-analysis.

## 1 Introduction

An object, e.g. an aircraft, is placed in a wind tunnel on supports. When there is no air flow in the tunnel, its position and orientation are well-defined. However, aerodynamic forces due to the flow of air through the tunnel may cause rotations or translations of the aircraft. In addition, the object is not completely rigid, but it may deform, e.g. its wings may bend up and down or twist. In order to be able to process the pressure and velocity data, one needs to know the position and orientation of the object as accurately as possible. For this purpose, a black and white CCD camera is available.

For the determination of the position and orientation of the object, well-defined reference points on the object are necessary. One might think of wing tips and the tail of an aircraft. However, we will assume that markers have been placed on the object, whose three-dimensional position on the model is known. We will also assume that we know which marker on the camera picture corresponds to which marker on the object. In this paper, we will describe two ways to determine the position and orientation of the model: an analytical method and a numerical least-squares approximation. Also we make some remarks about related questions.

## 2 Problem definition

A camera is placed at the origin. Its viewing direction is the positive $y$ axis. All coordinates are made dimensionless on the coordinate of the camera plane. The object in the wind tunnel is rotated and translated. So, a 3-dimensional point in the model reference frame is first rotated along the angles $\alpha, \beta$ and $\omega$, also known as pitch, yaw and roll which correspond to rotations along the $y, z, x$ axes, respectively. It is then translated by the vector $\mathbf{t} = (x_t, y_t, z_t)$. The total transformation from a vector $\mathbf{v}^o$ in object space to $\mathbf{v}^c$ in camera space thus becomes

$$\mathbf{v}^c = \mathbf{T}(\mathbf{v}^o) = \mathbf{t} + R_\omega R_\beta R_\alpha \mathbf{v}^o, \tag{1}$$

where the $R$-matrices [1] are rotation matrices for the angles $\alpha, \beta, \omega$, given by

$$R_\alpha = \begin{pmatrix} \cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) \end{pmatrix},$$

$$R_\beta = \begin{pmatrix} \cos(\beta) & -\sin(\beta) & 0 \\ \sin(\beta) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$R_\omega = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\omega) & -\sin(\omega) \\ 0 & \sin(\omega) & \cos(\omega) \end{pmatrix}. \tag{2}$$

A point $\mathbf{v}^c = (x, y, z)$ is projected on the $y = 1$ plane (equivalent to the camera space) according to

$$\mathbf{v}^p = \mathbf{P}(\mathbf{v}^c) = (p, q) = (\frac{x}{y}, \frac{z}{y}), \tag{3}$$

so the total projection operator becomes

$$\mathbf{v}^p = \mathbf{P}(\mathbf{T}(\mathbf{v}^o)). \tag{4}$$

A picture of the different types of coordinates is given in Figure 1. For a set of points on the same object the angles $\alpha$, $\beta$, $\omega$ and the translation vector $\mathbf{t}$ are the same (if the body is undeformed) or strictly related (if the body is allowed to deform in a particular way). Therefore, $\alpha$, $\beta$, $\omega$ and $\mathbf{t}$ can be found from the projection information of a large enough number of points.

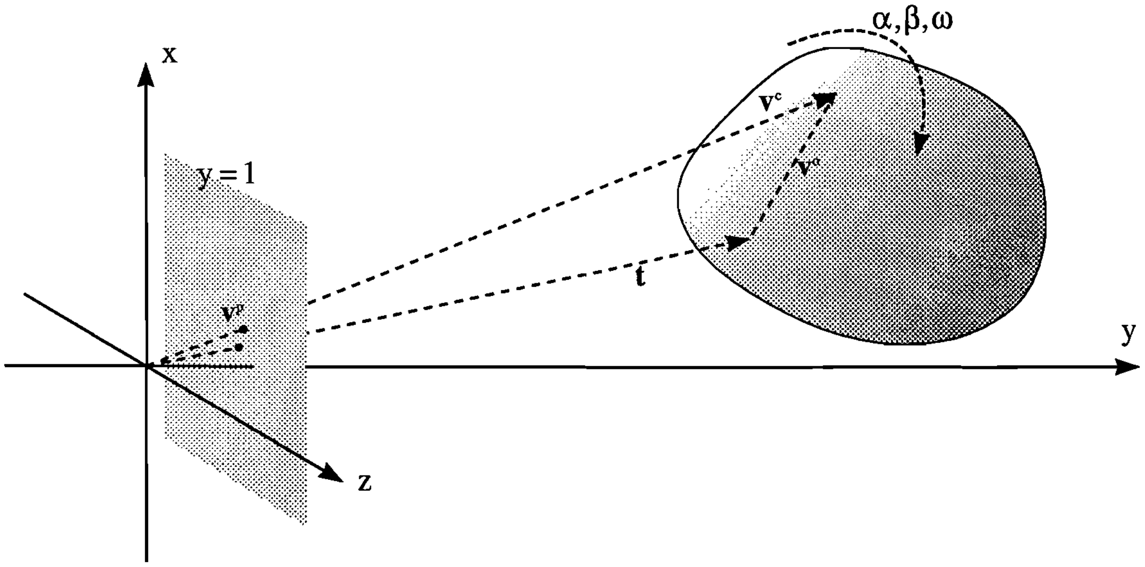

Figure 1: The different coordinates

# 3 Analytical approach

The total number of unknowns is 6, so a measurement of the three points $(p_1, q_1), (p_2, q_2)$ and $(p_3, q_3)$ should give a solution. However, this solution should be real and unique. We will investigate whether such a solution exists.

Since nothing is known a priori about the position and orientation of the object, the only information about the points is the distance between them in object space. Since the rotation and translation transformations are unitary, these distances remain the same in camera space. So we will use the distances to determine the 3-dimensional position of the points in camera space, after which the rotation and translation parameters can easily be deduced.

From the observation of the position $(p_i, q_i)$ on the camera of a point, it is clear that, in camera space, the point must lie on a line parameterised by $y_i$

$$\mathbf{v}_i^c = y_i(p_i, 1, q_i). \tag{5}$$

For convenience, we will introduce the dimensionless parameters $\lambda$ and $\mu$

$$\lambda := \frac{y_2}{y_1}, \qquad \mu := \frac{y_3}{y_1}. \tag{6}$$

Then the distances between the points can be expressed in $y_1, \lambda$ and $\mu$

$$\begin{aligned} d_{12} &= y_1\sqrt{(p_1 - \lambda p_2)^2 + (1-\lambda)^2 + (q_1 - \lambda q_2)^2} \\ d_{13} &= y_1\sqrt{(p_1 - \mu p_3)^2 + (1-\mu)^2 + (q_1 - \mu q_3)^2} \\ d_{23} &= y_1\sqrt{(\lambda p_2 - \mu p_3)^2 + (\lambda-\mu)^2 + (\lambda q_2 - \mu q_3)^2} \end{aligned} \tag{7}$$

By eliminating $y_1$ this can be reduced to the following two quadratic equations in $\lambda$ and $\mu$

$$\begin{aligned} &\frac{p_2^2 + q_2^2 + 1}{d_{12}^2}\lambda^2 - \frac{p_3^2 + q_3^2 + 1}{d_{13}^2}\mu^2 + \frac{p_1p_2 + q_1q_2 + 1}{d_{12}^2}\lambda + \cdots \\ &\qquad + \frac{p_1p_3 + q_1q_3 + 1}{d_{13}^2}\mu = -(p_1^2 + q_1^2 + 1)\left(\frac{1}{d_{12}^2} + \frac{1}{d_{13}^2}\right) \end{aligned} \tag{8}$$

$$\begin{aligned} &\frac{p_2^2 + q_2^2 + 1}{d_{23}^2}\lambda^2 - \frac{p_2p_3 + q_2q_3 + 1}{d_{23}^2}\lambda\mu + \frac{p_3^2 + q_3^2 + 1}{d_{23}^2}\left(1 - \frac{d_{23}^2}{d_{13}^2}\right)\mu^2 + \cdots \\ &\qquad + \frac{p_1p_3 + q_1q_3 + 1}{d_{13}^2}\mu = \frac{p_1^2 + q_1^2 + 1}{d_{13}^2}. \end{aligned} \tag{9}$$

These equations describe curves in the $\lambda$-$\mu$ plane. The shape of these curves depends on the parameters: for a general quadratic equation of the form $ax^2 + bxy + cy^2 + dx + ey = f$, the sign of the quantity $D = b^2 - 4ac$ determines whether the curve is hyperbolic, parabolic or elliptic in nature. If D is positive, the curve is a hyperbola, if D is zero, it is a parabola and if D is negative, the curve becomes an ellipse. It is clear that equation (8) always describes a hyperbola; for equation (9), this depends on the parameters, although we can say with certainty that it will be a hyperbola when $d_{23} \geq d_{13}$.

In general, these two equations (8) and (9) yield four (possibly degenerate) solutions. If there are complex solutions, they will always be in pairs of complex conjugates, since all coefficients of the equations are real. A real-world solution is real; there may be 0, 2 or 4 real solutions. This means that the solution of this problem with three markers is not unique.

We will clarify this with an example: Take a triangle with its corner points at $\mathbf{v}_1^c = (-1, 4, -1), \mathbf{v}_2^c = (-1, 4, 1)$ and $\mathbf{v}_3^c = (1, 4, 0)$. These will project at $\mathbf{v}_1^p = (-1/4, -1/4), \mathbf{v}_2^p = (-\frac{1}{4}, \frac{1}{4})$ and $\mathbf{v}_3^p = (\frac{1}{4}, 0)$. However, if $\mathbf{v}_3^c$ were located at $(\frac{13}{17}, \frac{52}{17}, 0)$, all sides of the triangle would still have the same lengths and $\mathbf{v}_3^p$ would still be the same. These solutions are two of the four possible solutions with this triangle and these projections.

In order to uniquely determine the solution, a fourth marker is necessary. Adding a fourth marker results in an over-determined problem. In the absence of errors this uniquely fixes the solution (when errors are present one has to minimize some measure of the total error, see the next section). The fourth marker should not be on an edge of the original triangle; then, the same problems can occur. In nearly all other cases, the fourth marker uniquely determines the solution. We will consider the following four markers in object space: $\mathbf{v}_1^o = (-1, 0.7, 0.3), \mathbf{v}_2^o = (0.5, -0.5, 0.6), \mathbf{v}_3^o = (0, 0.1, 0.6)$ and $\mathbf{v}_4^o = (-1, 0.3, 1.0)$. They are rotated along $\alpha = -0.3, \beta = 0.2$ and $\omega = 0.5$ and translated along $(0, 2, 0)$ into camera space. Two triangles are constructed from this; $\triangle_1$ consists of $\mathbf{v}_1^o, \mathbf{v}_2^o$ and $\mathbf{v}_3^o$, and $\triangle_2$ consists of $\mathbf{v}_1^o, \mathbf{v}_2^o$ and $\mathbf{v}_4^o$. The procedure described above is applied to determine the curves described by equations (7) and (8) for each triangle. For both triangles, the curves are hyperbolas. Figures 2 and 3 show the curves for respectively $\triangle_1$ and $\triangle_2$. Both triangles yield 2 solutions (cross-sections of the curves).

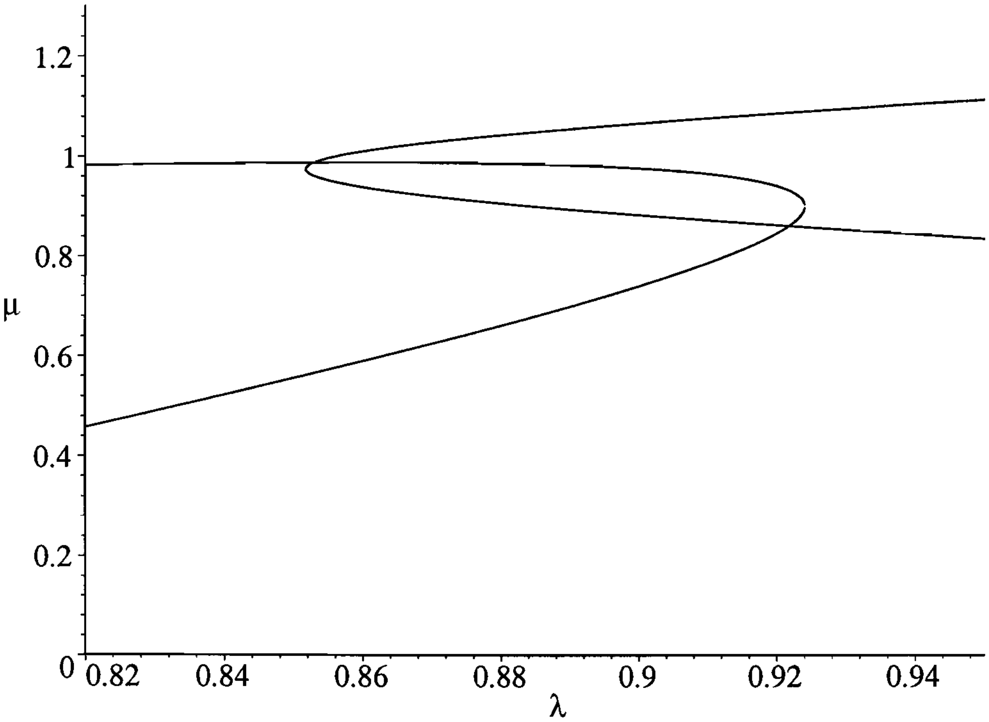

Figure 2: Relevant portion of curves of $\triangle_1$

Triangle $\triangle_1$ yields real solutions at $(\lambda, \mu) = (0.86925, 0.76122)$ and at $(0.77882, 0.98656)$; the solutions for $\triangle_2$ are at $(\lambda, \mu) = (0.77882, 1.13132)$ and $(1.12031, 0.52719)$. In both cases, $\lambda$ is defined as $y_2/y_1$, so the solutions for which $\lambda$ is equal are the solutions in which the calculated $\mathbf{v}_2^c$ is at the same position in both triangles. The calculated $\mathbf{v}_3^c$ is also the same in both triangles.

When the positions of the markers in camera space are known, the corresponding rotation matrix and translation can be easily calculated, and the rotation angles $\alpha, \beta, \omega$ and the translation vector $\mathbf{t}$ are equal to the real values (up to $10^{-8}$).

# 4 The method of least squares

The method of (nonlinear) least squares [2] can be used to reconstruct from observed data one or more parameters, in particular when there is more data than parameters. Let $F$ be a map that maps the parameters $(\mathbf{t}, \alpha, \beta, \omega)$, and the marker positions in object coordinates $\mathbf{v}_i$ to the camera

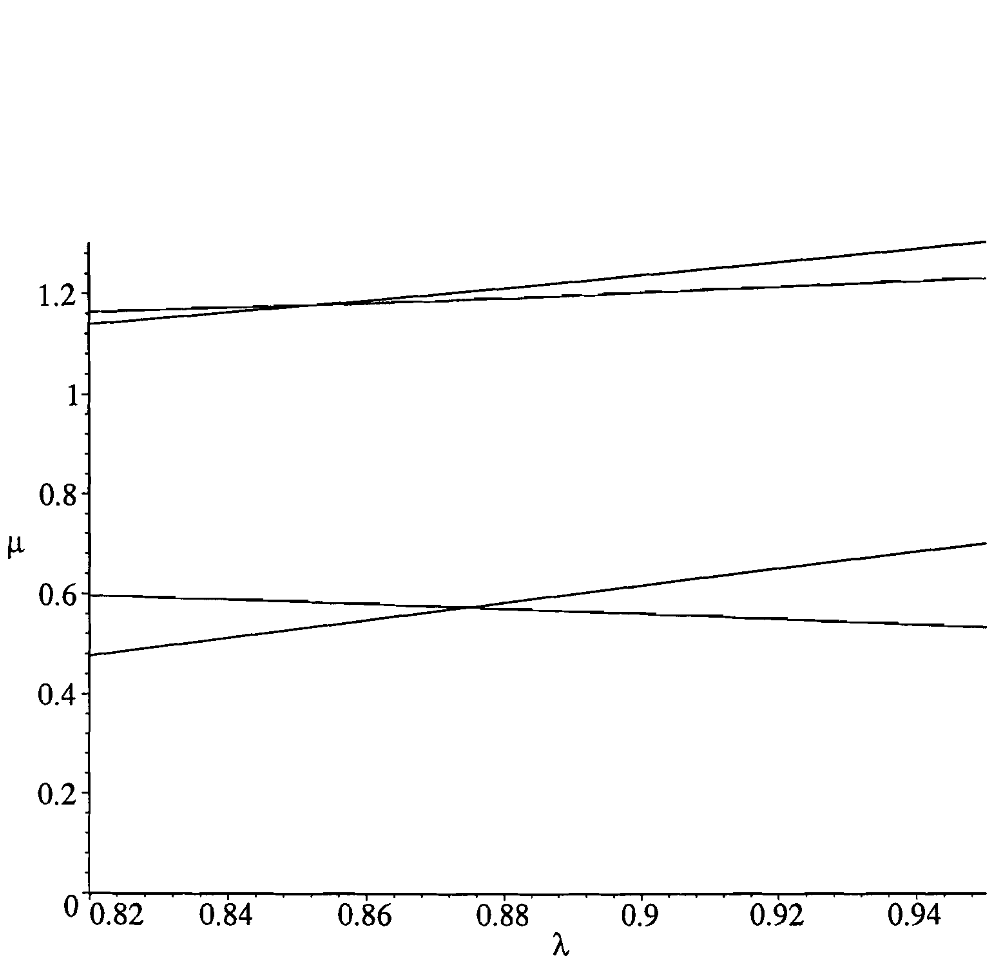

Figure 3: Relevant portion of curves of $\Delta_2$

points $(p_i, q_i)$

$$
\begin{aligned}
(p_i, q_i) &= F(\mathbf{t}, \alpha, \beta, \omega; \mathbf{v}_i) + \text{error} \\
&:= P(\mathbf{t} + R_\omega R_\beta R_\alpha \mathbf{v}_i) + \text{error}.
\end{aligned} \tag{10}
$$

Suppose there are $k$ markers. Let the vector $\mathbf{d} = \{(p_1, q_1), \dots, (p_k, q_k)\}$ be the data. We now try to find $(\mathbf{t}, \alpha, \beta, \omega)$ that minimize the squared error, i.e. the squared difference of observed and modelled data given by

$$
\begin{aligned}
&E((p_1, q_1), \dots, (p_k, q_k); \mathbf{t}, \alpha, \beta, \omega; \mathbf{v}_1, \dots, \mathbf{v}_k) \\
&\quad := \sum_{i=1}^{k} [(p_i - p(\mathbf{t}, \alpha, \beta, \omega; \mathbf{v}_i))^2 + (q_i - q(\mathbf{t}, \alpha, \beta, \omega; \mathbf{v}_i))^2].
\end{aligned} \tag{11}
$$

Note that the least squares method treats all data on equal footing, unlike the method of the previous section where we solved using three data points and then used the fourth point to determine which of the solutions was the correct solution. Therefore the least squares method is probably a better way of dealing with the data.

The method was invented by the famous German mathematician C.F. Gauss who used it to do a geodesic survey with a large precision. He knew that doing just enough measurements would result in an error that was far too large, so he had many measurements done and using the method least squares the individual errors would average out.

In order to minimize this error function $E$, one selects a suitable starting point and follows the "path of steepest descent" down to a minimum. In general such a function may have many

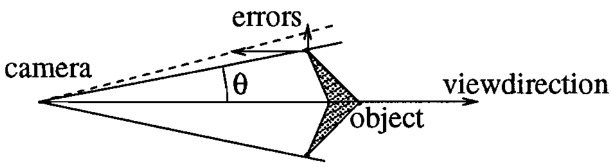

Figure 4: The ratio of the errors depends on the view angle $\theta$.

minima, and this process will converge to a minimum that is close to the starting point. If we have a good starting point it is a very efficient method. In our case we take many images per second, and the parameters from the previous image will most likely be a good starting point. Therefore we think it is very important to use the information from the previous image.

Using the mathematical theory one should investigate when the least squares method converges to the correct minimum in a reasonable time. This depends on the behavior of the function $E$. It is also possible to obtain an estimate for the errors in the resulting position and orientation. We have not worked this out.

Using standard mathematical software such as Mathematica, Maple or Matlab it is not difficult to implement the method. We used Mathematica to calculate a few examples. We first did experiments with 3 markers. In this case the number of data equals the number of unknowns (so this is not really least squares, but it is equation solving and the minimum value of the error should be zero). We have seen above that there are in general four possible solutions in this case. If these are well separated, and the starting point is sufficiently close to the correct solution then the algorithm converges indeed to the correct solution. On the other hand, when the starting point is too far away from the correct solution, we obtain a wrong answer. Also there were cases where the convergence was relatively slow, which probably corresponded to either the situation of two minima close to each other, or a degenerate minimum.

After that we did experiments with four markers, where the fourth marker was not in the plane determined by the other three. In the examples we did, the algorithm converged to the correct solution.

It is possible to estimate the error in the result due to error in the observed data, simply by perturbing the correct data with some Gaussian error, and then comparing the parameters obtained from this data with the correct parameters. In fact we may take a set of perturbed data, and then look at the set of parameters that is obtained, and the shape of this set indicates what directions are sensitive to errors.

We used this procedure to compare the error in the camera direction with the error in the directions perpendicular to the camera direction. We expect that the ratio of error in camera direction and perpendicular to it is approximately the tangent of the view angle $\theta$ (see Figure 4) To test this we put a triangle with coordinates $(-1,0,0),(0,0,2),(1,0,0)$ at position $(0,2,0)$ resp. $(0,20,0)$. The error clouds due to a Gaussian perturbation of the data with $\sigma = 0.02$ resp. $\sigma = 0.002$ are given in Figure 5. In the first picture the ratio of errors in $x,y$ coordinates is approximately 1, in the second it is approximately 10, in both cases it is approximately $2\tan\theta$.

## 5 Extensions of this work

When defining the problem above there are a number of aspects that we disregarded.

When the object is deformed it is not clear what should be called the position and orientation of the object. This introduces an error. Also we may want to determine the amount of deformation of the object. We suggest the following approach to this problem. The marker position in the object frame will now depend on the deformation. We assume that the position of the marker can be parametrized by some deformation parameter $\lambda$,

$$\mathbf{v}^o = \mathbf{v}^o(\lambda),$$

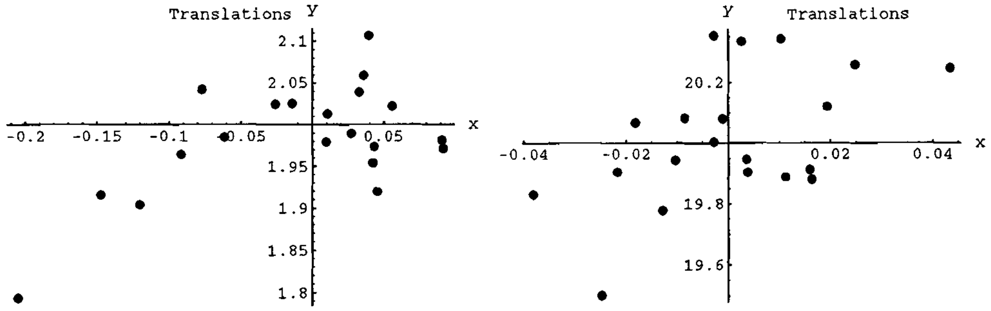

Figure 5: Reconstructed $x, y$ translation parameters with Gaussian errors in the data.

In general when several types of deformation are possible we need $m$ parameters $\lambda = (\lambda_1, \ldots, \lambda_m)$. Now the locations in the camera view depend also on the parameter $\lambda$

$$(p_i, q_i) = F(\mathrm{t}, \alpha, \beta, \omega; v_i(\lambda)).$$

Using the least squares algorithm we can also determine $\lambda$. This idea has been tested using our Mathematica notebook, in the situation of one point being displaced in $z$-direction. In the examples with four marker points, one of which was displaced due to assumed deformation, the displacement could be reconstructed, as long as it is not in camera direction (as one would expect).

Another problem is to match the observed markers with the markers on the object. This is complicated by the fact that sometimes markers may be out of the view of the camera, for instance because they are on the backside of the object or because they are hidden by an other part of the object. In addition, there may be spurious or undetected markers. We suggest to use the information of the previous image to solve this problem. From the previous image the location of the different markers is approximately known, and from this information one can obtain information about the matching of the markers. Also one can determine which markers are likely to disappear out of the view.

# 6 Other suggestions

Finally we suggest a few things that could be important in extending this work. One possible extension is the use of multiple cameras.

Secondly, we mention the detection of edges and corners. If that could be automated then there would be no need to attach any markers to the object.

A third thing to do would be investigating the literature for results on similar problems. We could think of geodesy, remote sensing, motion capturing.

# References

[1] O. Bottema, *Theoretische Mechanica*, Epsilon Uitgaven, Utrecht, 1985.

[2] J. Stoer and R. Bulirsch, *Numerical Analysis*, Springer, New York, 1980.

# Cache as ca$h can

W.J. Grootjans, M. Hochstenbach, J. Hurink,
W. Kern, M. Luczak, Q. Puite, J.A.C. Resing, F.C.R. Spieksma

**Abstract**

In this contribution several caching strategies for the World Wide Web are studied. Special attention is paid to the so-called proxy placement, i.e. placing of caches on carefully selected nodes in the network near to the end users. Using both a deterministic and a stochastic approach, algorithms are developed for calculating the allocation and sizes of caches with the aim to enhance the performance of the network. Under the restriction of fixed budget it is also indicated how both approaches can be combined.

**Keywords**

Cache assignment, World Wide Web, Allocation, Proxy placement.

## 1 Introduction

The World Wide Web (WWW) has experienced continuous, exponential growth since its inception in the beginning of the 90's. This has led to a considerable increase in the amount of traffic over the internet. As a consequence, Web users nowadays can experience large waiting times (latencies) due to network congestion and/or server overloading. Moreover, if current predictions concerning usage of the Web come true, this performance issue will become even more important in the near future.

A way to improve the performance of the Web is caching (as witnessed by for instance [2], [11]). Caching copies of popular objects closer to the user is an important way of improving the network's performance. Indeed, the two main potential benefits of caching are: reduction of latencies experienced by the users, and saving of bandwidth, due to a decrease of network traffic. In order to realize these potential benefits, at least two (related) problems have to be dealt with:

- how to operate a cache. There is a sizable literature devoted to caching strategies to improve Web performance; see, for instance, [1] and [9] where algorithms generalizing the well-known LRU algorithm are proposed.
- where to install cache. Different caching options are possible: on the one hand, objects can be stored at the user's browser, which gives the possibility to make use of the user's individual characteristics (client caching, see for instance [1] and [4]); on the other hand objects can be stored in the cache of the Web server (see [8]). In between these options, there is the option of using *proxies*, that is, to install specialized servers at various points in the network (first proposed by [6]; see also [3]). Typically, such an approach is attractive when an organization (like a company or a university) is responsible for (a part of) a network ([7]).

This paper deals with the latter subject called proxy placement in [10]: given a network with capacitated edges, external demand (request rates for objects and their sizes), costs for installing a proxy and a budget, we develop a heuristic method to decide where to install proxy caches in the network and what the sizes of these caches should be. The heuristic attempts to minimize a function of the waiting times in the network, for instance, the average waiting time. We assume that only passive caching is used, i.e., features like pre-fetching and pre-loading are excluded.

The rest of this paper is organized as follows. Section 2 describes the problem and introduces some terminology. In Section 3 we propose an algorithm that suggests a proxy placement to minimize waiting times. Finally, Section 4 analyzes some stochastic aspects of the problem.

## 2 Problem description, notation and terminology

The input for our problem is as follows:

1. An *infrastructure* $\mathcal{T} = (\mathcal{V} \cup \{\infty\}, \mathcal{E})$ is a rooted tree ($\infty$ standing for the root), such that the root has exactly one child. The root represents the outside world and the *inner vertices* (elements of $\mathcal{V}$) represent servers. The *edges* (elements of $\mathcal{E}$) are directed in direction of the root and represent connections between the servers. The relation "$i'$ is a child of $i$" generates a partial order $\preccurlyeq$ ("descendant of") with top $\infty$ (i.e. $\forall i \in \mathcal{V} : i \preccurlyeq \infty$).

   Observe that the inner nodes are in 1-1-correspondence with the edges, each inner node being a child of another node via the corresponding edge. Let $H$ denote the height of the tree and $M$ the maximal number of children per node. Typical values for $H$ and $M$ are 5 and 100 respectively.

2. The files $1, \ldots, N$ that are requested at the servers have sizes $s_1, \ldots, s_N$, are located ouside the infrastructure, and can be achieved only via the root.

3. Let $\lambda_{i,j}$ denote the *frequency* of requests for file $j$ at (inner) node $i$ (in terms of number of requests per time unit). The following quantities are closely related to these frequencies: let

   $$\lambda_i := \sum_{j=1}^{N} \lambda_{i,j}$$

   denote the *total frequency* of requests at node $i$. Then

   $$p_{i,j} := \frac{\lambda_{i,j}}{\lambda_i}$$

   denotes the *relative frequency* of the requests for file $j$ at server $i$. That is, for every $i$ the function $j \mapsto p_{i,j}$ is a discrete distribution, $\sum_{j=1}^{N} p_{i,j} = 1$.

   The *demand* (data per time unit) generated by requests for file $j$ at server $i$ equals

   $$\kappa_{i,j} := \lambda_{i,j} s_j.$$

   The *total demand* caused at server $i$ is given by

   $$\kappa_i := \sum_{j=1}^{N} \kappa_{i,j} = \lambda_i \sum_{j=1}^{N} p_{i,j} s_j.$$

4. Each edge $e \in \mathcal{E}$ has a *capacity* $c_e \geq 0$: the maximal flow (in terms of data per time unit) through the edge.

5. The costs to place a proxy in a node $i$ are a linear function of the size of the cache $y$. If $a$ denotes the fixed and $b$ the variable costs, we can write the costs as

   $$k(y) = \begin{cases} a + by & (y > 0) \\ 0 & (y = 0) \end{cases}$$

6. We have a total budget $B > 0$ to purchase proxies.

It is important to realize that given this input, any decision concerning the location, size and a local caching strategy for each proxy determines, in a unique fashion, flows in the network $\mathcal{T}$ (assuming that requests are served from proxies "up the tree" or from $\infty$ in case there are no proxies up the tree that contain the specific request). We will use variables $x_e$, $e \in \mathcal{E}$ to denote these flows. In particular, if no proxies at all are installed in $\mathcal{T}$, one can compute that the edge flows, denoted by $x_e^0$ in this case, are equal to $\sum_{i' \preccurlyeq i} \kappa_{i'}$ where $i$ is the node directly under edge $e$. Let us now explicitly describe the assumptions that we use in our model:

- as mentioned above, requests are served by the closest proxy up the tree that contains the requested object. This assumption is reasonable in practice.
- in Section 3 we assume a *static* strategy as a local caching strategy, that is a set of files is chosen to remain in the cache permanently. Obviously, this is a crude simplification of reality, where LRU type of caching strategies are common. However, in this section we are primarily interested in proxy placement and their corresponding sizes; the specific local caching strategy is of minor importance in our setting which justifies this assumption. Afterwards, in Section 4 the results are analysed on the base of a LRU caching strategy.
- to be able to compute a waiting time for each edge $e \in \mathcal{E}$ (denoted by $w_e$), we rely on the following relation between waiting time $w_e$, (given) capacity $c_e$ and flow $x_e$:

  $$w_e = C \times \left(1 - \frac{x_e}{c_e}\right)^{-1} \qquad \text{for some constant } C.$$

- we assume that the objective function that we want to minimize, say $h$, can be expressed in terms of the waiting times $w_e$. For example, suppose one would like to minimize the largest waiting time experienced by some user in the network. This can be formulated as follows: let $\mathcal{P}$ denote the set of all paths from $\infty$ to any inner node, then the objective function is given by

  $$h = \max_{p \in \mathcal{P}} \sum_{e \in p} w_e.$$

  Alternatively, the average waiting time in the tree over all possible paths can be formulated as

  $$\frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \sum_{e \in p} w_e = \frac{1}{|\mathcal{P}|} \sum_{e \in \mathcal{E}} w_e \left|\{p \in \mathcal{P} \mid e \in p\}\right| = \frac{1}{|\mathcal{V}|} \sum_{e \in \mathcal{E}} n_e w_e,$$

  where $n_e$ is the number of nodes in the subtree under $e$.

Concluding, a solution to our problem specifies for each node in the tree whether or not a proxy is installed, and, if so, it specifies its corresponding size. In addition, specific files are suggested to be stored in each proxy. All this is done while attempting to minimize (a function of) the waiting times.

# 3 The cache assignment

In this section we will consider the problem of determining the nodes in the tree where cache will be assigned and the files which will be stored in these caches. The goal is to achieve a comfortable situation for the users of the network. Due to the assumptions made in the previous section, the quality of the solutions depends on the waiting times and, therefore, on the loads in the edges.

Our heuristic algorithm to solve the problem consists of 2 major steps that are performed iteratively. In Step 1, we specify *upper bounds* $u_e$ on the loads for each $e \in \mathcal{E}$. Next, in Step 2 we (heuristically) decide whether a proxy placement exists such that $x_e \leq u_e$ for each $e \in \mathcal{E}$, and such that the total expenses remain within budget $B$. If the answer is yes, we update the upper bounds $u_e$ in such a way that they correspond to a more comfortable situation and iterate, otherwise we either stop, or relax the current upper bounds. Subsection 3.1 deals with Step 1 and the updating of the upper bounds and Subsection 3.2 describes Step 2.

### 3.1 Step 1: computing and updating upper bounds $u_e$

The basic structure of the algorithm is visualized in Figure 1. It consists of an initialization of the bounds and a loop process in which the upper bounds are updated according to the the procedure which will be explained in the next section. In the following some remarks how these steps may be realized are given:

- To start, the algorithm needs an initial set of total edge flows for all edges. We propose two ways to find this set of flows: when the instance under consideration corresponds to an existing network, one can use the current situation as a starting point. More specifically, the current proxy placement and the current flows can be used as input for the algorithm. Another possible way to get an initial flow is as follows: specify a maximal waiting time for each edge $e \in \mathcal{E}$: $w_e^*$. Now $w_e \leq w_e^*$ is equivalent to

  $$x_e \leq \left(1 - \frac{1}{w_e^*}\right) c_e =: u_e.$$

  The corresponding flows can be used as input for the algorithm.

- The upper bounds are updated to the current total edge flows and a factor times the gradient of the objective function is subtracted. This means that the upper bound for each total edge flow is reduced proportional to the rate of descent of the objective function with respect to that total edge flow, which is symbolically denoted by $\nabla h(\mathbf{x})$. This gradient is evaluated for the current total edge flows. Notice that we rely here on the assumption that we are able to compute this gradient (cf. the choices of $h$ mentioned in Section 2).

- The question whether or not an allocation with $\mathbf{x} \leq \mathbf{u}$ (for all components) exists, can be answered by the "inner loop" which yields either the answer *no* or the answer *yes* and a set of total edge flows $\mathbf{x}$.

  If the answer from Step 2 is negative then we go back a few steps in the algorithm and decrease the upper bounds less than we did initially, or we have found a solution that we consider satisfactory and stop.

  If the answer from Step 2 is positive then the new set of total edge flows becomes the set of upper bounds and the algorithm reiterates.

- The value of $\alpha_0$ and the way $\alpha$ is decreased will have to be looked at using an implementation. At this time we cannot say anything sane on these matters.

### 3.2 Step 2: proxy placement for given upper bounds $u_e$

Given upper bounds $u_e$, we determine whether we can find flows $x_e \leq u_e$ by allocating a total amount of cache of cost $\leq B$. We divide this problem into three subproblems:

P1) Determine in which nodes a proxy is installed;

P2) Determine how much cache is installed in these nodes;

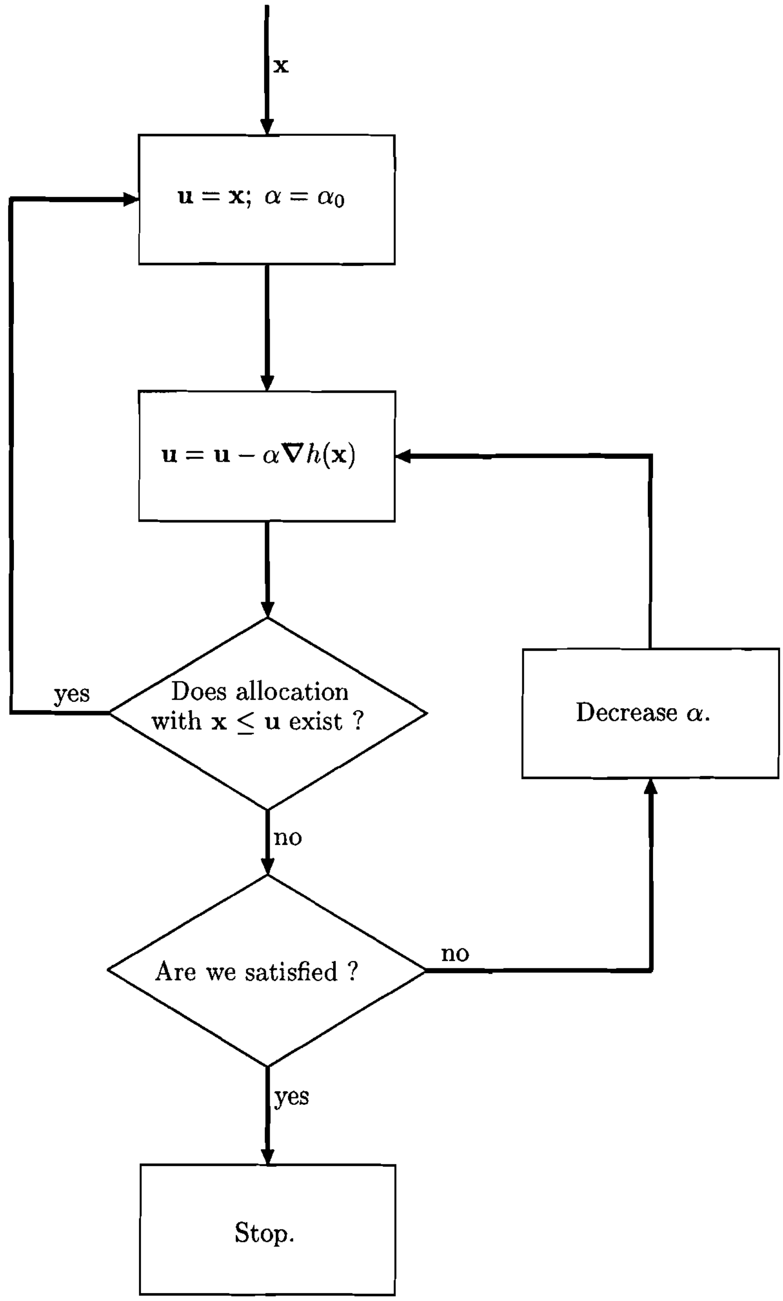
x
$\mathbf{u} = \mathbf{x};\ \alpha = \alpha_0$
$\mathbf{u} = \mathbf{u} - \alpha \nabla h(\mathbf{x})$
yes
Does allocation with $\mathbf{x} \leq \mathbf{u}$ exist ?
Decrease $\alpha$.
no
no
Are we satisfied ?
yes
Stop.

Figure 1: Flowchart of the updating of the bounds.

P3) Determine which files are stored in these proxies.

We solve these three subproblems by first considering P1, and next P2 and P3 simultaneously. Notice that Step 2 either outputs "yes" with an accompanying flow $x$ or "no" meaning: no flow $x$ with $x_e \leq u_e$ with expenses $\leq B$ is found.

### 3.2.1 Determine nodes in which a proxy is installed

In this subsection we describe an algorithm that determines in which nodes of $\mathcal{T}$ a proxy is installed. In fact, this algorithm determines the *minimal number* of proxies and possible locations for them that are necessary to achieve flows $x_e \leq u_e$ by a straightforward "bottom up" search in linear time. However, although our result is minimal in the mentioned sense, it may not be optimal concerning the complete problem.

To describe the algorithm, first we need some additional notation: If $u_e$ is an upper bound we impose on the flow $x_e$ then the overflow $o_e$ with respect to $x$ is defined as:

$$o_e = \begin{cases} x_e - u_e & (x_e > u_e) \\ 0 & (x_e \leq u_e) \end{cases}$$

If $o_e > 0$ we call $e$ an *overflow edge* with respect to flow $x$.

The algorithm works as follows. Let $e_0$ be an overflow edge ($o_{e_0} > 0$) with respect to flow $x^0$ without any overflow edges below $e_0$. Then we place a proxy in the corresponding node $i_0$ and compute a new flow $x$, *assuming* $x_{e_0}$ *vanishes: the total request of the corresponding subtree becomes zero ("complete caching").* Observe that the flow only changes in edges along the path from $i_0$ to $\infty$: for each of these edges $e$ we set $x_e := x_e - x^0_{e_0}$. Next, we find a new overflow edge with respect to this updated flow and repeat until no overflow edges exist in $\mathcal{T}$.

**Claim:** *Assuming the budget constraint is not violated, this algorithm determines a minimal number of nodes where a proxy must be installed in order to be able to output "yes".*

**Argument:** Consider an overflow edge $e$ which has no overflow edge below it. Since the flow in this edge has to be reduced to get a feasible solution, we have to place at least one proxy below $e$. The maximal reduction of the load in $e$ by placing proxies below $e$ is equal to the current load of $e$. However, since the placement of a proxy of arbitrary large size in the node corresponding to $e$ leads to this reduction, it will be optimal to place the proxy there. These observations imply that for each overflow edge that has no overflow edge below it in $\mathcal{T}$, a proxy must be installed at the corresponding node. Moreover, it follows from our approach that the demand from nodes where a proxy is placed vanishes. The proof of the claim now follows by induction. □

Remarks:

- This algorithm determines the actual nodes where a proxy is located, which gives us a value of the fixed costs of the cache assignment. Indeed, if this sum of fixed costs exceeds budget $B$, we stop and output: no feasible solution found.
- Moreover, it gives lower bounds on the cache sizes. Indeed, putting file $j$ of size $s_j$ in the cache of node $i_0$ decreases $x_{e_0}$ by

  $$s_j \sum_{\text{relevant } i \preccurlyeq i_0} \lambda_{i,j},$$

  which is a decrease of $\widehat{\lambda}_{i_0,j} := \sum_{\text{relevant } i \preccurlyeq i_0} \lambda_{i,j}$ per unit cache. (Relevant $i \preccurlyeq i_0$ means: $i$ below or equal to $i_0$ such that there is no proxy in between $i$ and $i_0$.) Let $\tau_{i_0}$ be an enumeration of the files $j$ in order of decreasing $\widehat{\lambda}_{i_0,j}$. Then the minimal needed cache size equals $\sum_{k=1}^{d-1} s_{\tau_{i_0}(k)}$, where $d$ is minimal such that $\sum_{k=1}^{d} \widehat{\lambda}_{i_0,\tau_{i_0}(k)} s_{\tau_{i_0}(k)} > o_{e_0}$. (Without complete caching below $i_0$ more $i$ may become relevant (probably depending on $j$), and one easily verifies the minimal needed cache size in node $i_0$ only increases.)

- The given algorithm may be modified by assuming that the placement of a cache at a node does not vanish the complete flow, but only a certain percentage $x$ (this seems to be more realistic). In this case we will loose the 'minimality property' but it may be easier to solve Steps P2) and P3).

### 3.2.2 Determine the sizes of the proxies and the stored files

First, let us deal with the complexity of these subproblems.

**Claim:** Given the nodes where proxies are installed, computing the minimum total size necessary to achieve $x_e \leq u_e$ is NP-hard.

**Argument:** We will prove the claim by a reduction from the partitioning problem. Let $n$ numbers $a_1, \ldots, a_n$ with $\sum_{i=1}^{n} a_i = 2b$ be given. The partition problem consist of answering the question whether or not a partition of $\{1, \ldots, n\}$ into two subsets $S_1, S_2$ with $\sum_{i \in S_1} a_i = \sum_{i \in S_2} a_i = b$ exists. We may reduce an instance of the partition problem to the following cache assignment problem: Given are $n$ leaves $1, \ldots, n$ which are all connected to a source $s$ and this source $s$ is connected to the root. Leave $i$ requests only for a file $f_i$ of size $a_i$ and source $s$ has no request. For the edge $(s, \infty)$ we define an upper bound $u = b$ and all other edges have upper bounds which are large enough. Furthermore, cache may only be installed at the source $s$. It is straightforward to see that it is possible to achieve a feasible solution with a total amount of cache equals $b$ if and only if the partition problem has a feasible solution. □

Thus, problem P2) is unlikely to be solvable exactly by a polynomial time algorithm.

In view of the result above, we will solve P2) & P3) simultaneously by a greedy heuristic. Generally stated our heuristic works as follows. Put files in cache, one at a time, so as to maximize the relative total overflow reduction in each step. Proceed greedily until the overflow on each edge in $\mathcal{T}$ is reduced to 0. This implies a cache size allocation. If this allocation has costs exceeding budget $B$ we return "no", otherwise we return "yes" with the corresponding flow $x$.

A more detailed description is as follows. Recall that an overflow edge is one with $o_e > 0$. The relative overflow reduction $r_{i_0,j}$ is the total reduction of overflow caused by putting one unit size of file $j$ in the cache at node $i_0$. This depends on both the frequency $\widehat{\lambda}_{i_0,j}$ of requests on file $j$ that arrive in node $i_0$, and the multiplicity $\mu_{i_0,j}$ counting the number of overflow edges above node $i_0$ up to the next proxy containing $j$. More precisely, putting file $j$ (of size $s_j$) in cache at node $i_0$ reduces total overflow with $r_{i_0,j} = \mu_{i_0,j}\widehat{\lambda}_{i_0,j}$.

Thus, in each step of the algorithm, the current situation is given by:

- The set of files cached per proxy so far;
- The set of overflow edges;
- The frequencies $\widehat{\lambda}_{i_0,j}$ of file $j$ at node $i_0$:

$$\widehat{\lambda}_{i_0,j} := \sum_{\textbf{relevant } i \preccurlyeq i_0} \lambda_{i,j},$$

  where for a fixed file $j$, the summation is over those nodes $i$ below or equal to $i_0$ such that $j$ is not cached in between $i$ and $i_0$;

- The relative overflow reductions $r_{i_0,j}$ of putting file $j$ in the proxy at node $i_0$:

$$r_{i_0,j} = \mu_{i_0,j}\widehat{\lambda}_{i_0,j}.$$

Now we can identify a node $i$ and a file $j$ for which $r_{i,j}$ is maximal, store this file $j$ in the cache at node $i$ and iterate.

Finally, the amount of cache in each proxy $i$ is computed as

$$y_i = \sum_{j \text{ cached in } i} s_j.$$

Observe that only here the file sizes come in. Finally, our algorithm returns the total costs

$$\sum_{\text{proxies } i} k(y_i).$$

Of course, as the number $N$ of files involved can be a fairly high number (depending upon the specific situation) this algorithm should be carefully implemented. Let us now suggest an efficient implementation of an updating step in the algorithm. Recall that $\mu_{i_0,j}$ is the number of overflow edges in between node $i_0$ and the first node above $i_0$ where $j$ is cached. We therefore consider two files at node $i_0$ *equivalent* if the first node on the path from $i_0$ to the root, where these files are cached, is the same. For each $i_0$, we order equivalence classes into lists according to decreasing frequencies $\widehat{\lambda}_{i_0,j}$. Note that ($i_0$-)equivalent files $j_1, j_2$ have the same multiplicity: $\mu_{i_0,j_1} = \mu_{i_0,j_2}$. After we put file $j$ in proxy at node $i_0$, the update procedure now consists of:

- Finding the right equivalence class for file $j$ in nodes below node $i_0$;
- updating frequencies of $j$ at certain nodes above $i_0$;
- in case for some original overflow edge $e$ the overflow reduces to 0, changing the multiplicity $\mu_{i,j}$ on those equivalence classes to whose multiplicity $e$ contributed before.

To illustrate the updating, we now provide an example.

**Example**: *Suppose we have the following current situation in node $E$ of the depicted infrastructure of Fig. 2.*

| $\mu_{E,j}$ | *contributing overflow edges* | *nearest proxy containing $j$* | $j$ *(in order of decreasing $\widehat{\lambda}_{E,j}$)* |
|---|---|---|---|
| *4* | $A', (B'), C', D', E'$ | $\infty$ | *permutation of* $[6], \ldots, [N]$, *say*<br>[10] *(28)* ; [6] *(23)* ; ... |
| *2* | $D', E'$ | $C$ | [1] *(60)* ; [5] *(50)* |
| *1* | $E'$ | $D$ | [2] *(100)* |
| *0* | *none* | $E$ | [4] *(0)* ; [3] *(0)* |

*Then*

$$\begin{aligned} r_{E,[10]} &= 4 \cdot 28 = 112, \\ r_{E,[1]} &= 2 \cdot 60 = 120, \\ r_{E,[2]} &= 1 \cdot 100 = 100 \textit{ and} \\ r_{E,[4]} &= 0 \cdot 0 = 0, \end{aligned}$$

*so* [1] *is a candidate to be put in stack at node $E$. However, there may be a proxy $i_0$ with even bigger maximal $r_{i_0,j}$.*

- *Suppose $i_0 = D$ has biggest maximal $r_{i_0,j}$, viz. for $j =$ [5](130) (with $\mu_{D,[5]} = 1$, whence $r_{D,[5]} = 130$). Putting* [5] *in the proxy at $D$ has the following effect on the lists in node $E$ (below $i_0 = D$), assuming — say — that $A'$ and $D'$ become overflow free:*

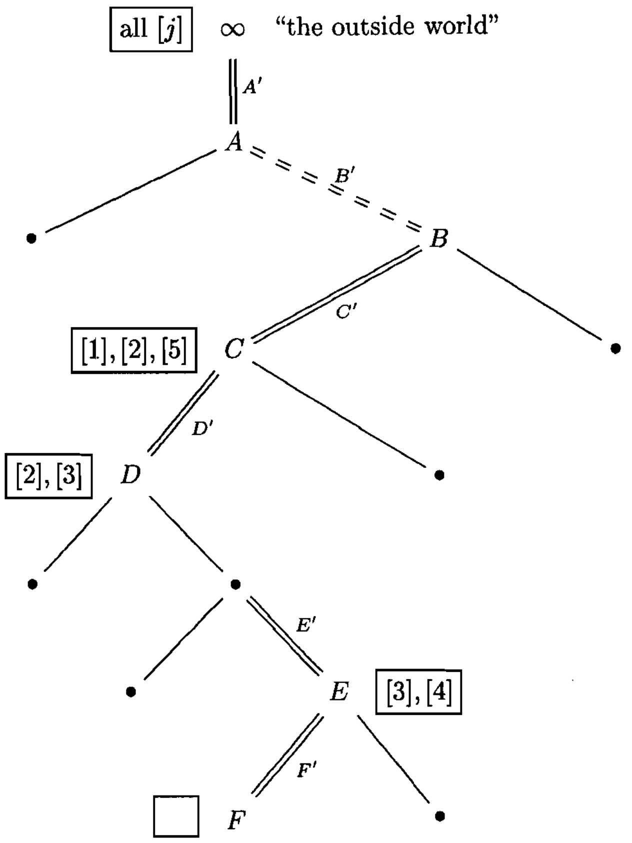

Figure 2: An infrastructure with original overflow edges $A', B', C', D', E', F'$, current overflow edges $A', C', D', E', F'$, proxies installed in $C, D, E, F$, currently containing the depictured files $[j]$.

| $\mu_{E,j}$ | *contributing overflow edges* | *nearest proxy containing j* | *j (in order of decreasing* $\widehat{\lambda}_{E,j}$*)* |
|---|---|---|---|
| *2* | $(A'), (B'), C', (D'), E'$ | ∞ | [10] *(28)* ; [6] *(23)* ; ... |
| *1* | $(D'), E'$ | *C* | [1] *(60)* |
| *1* | $E'$ | *D* | [2] *(100)* ; [5] *(50)* |
| *0* | *none* | *E* | [4] *(0)* ; [3] *(0)* |

- *Suppose* $i_0 = F$ *has biggest maximal* $r_{i_0,j}$*, viz. for* $j = [10]$*(25) (with* $\mu_{F,[10]} = 5$*, whence* $r_{F,[10]} = 125$*). Putting* [10] *in the proxy at F has the following effect on the lists in node E (above* $i_0 = F$*), assuming — say — that* $A'$ *and* $D'$ *become overflow free:*

| $\mu_{E,j}$ | *contributing overflow edges* | *nearest proxy containing j* | *j (in order of decreasing* $\widehat{\lambda}_{E,j}$*)* |
|---|---|---|---|
| *2* | $(A'), (B'), C', (D'), E'$ | $\infty$ | [6] *(23)* ; ... ; $\underline{\underline{[10]\ \ (3)}}$ ; ... |
| *1* | $(D'), E'$ | $C$ | [1] *(60)* ; [5] *(50)* |
| *1* | $E'$ | $D$ | [2] *(100)* |
| *0* | *none* | $E$ | [4] *(0)* ; [3] *(0)* |

# 4 Stochastic Analysis

So far we analyzed the caching problem using deterministic methods. We tackled the question *where* to cache specific files. In practice, however, no local caching strategy would cache *specific* files. Instead, one often uses the so-called Least Recently Used (LRU) strategy, which operates as follows:

- whenever a request arrives for a file that is not in cache, this file is cached,
- whenever the total size of the files in cache exceeds the cache capacity, the least recently used files are dropped.

In the following subsections we present a stochastic analysis of the caching problem that takes into account the LRU strategy. By this analysis we will estimate the expected loads on the edges of the network that result from a given assignment of cache to the nodes. Thus, using these results we may get a better view on the quality of the solutions achieved by the methods presented in the previous section.

## 4.1 Problem Formulation

As in the deterministic analysis, we consider the web-caching problem on a infrastructure $T = (V, E)$ with node (vertex) set $V$ and edge set $E$. We assume there is a fixed number $N$ of files in the network that can be requested by users, and associated with each node of the tree is a *fixed* number of files that can be stored simultaneously in its cache. Note that, for convenience, we neglect the fact that different files can have different sizes.

We are given the following parameters:

- $\lambda_{i,j}$, the frequency of arrivals of external requests for file $j$ at node $i$;
- $\lambda_i$, total frequency of external request arrivals at node $i$ (thus $\sum_{j=1}^{N} \lambda_{i,j} = \lambda_i$);
- $p_{i,j} = \frac{\lambda_{i,j}}{\lambda_i}$, the probability that a request arriving at node $i$ is for file $j$;
- $M_i$, the number of files that can be simultaneously stored at node $i$.

The aim is to determine the *expected* load $x_e$ on edge $e$, for all $e \in E$ when the LRU strategy is used. This is done by calculations that start at the leaves of the tree and successively working our way up to the root of the tree. For each node $i$, we first compute the total arrival rate $\bar{\lambda}_{i,j}$ of requests for file $j$ at $i$, which also includes the requests received from all descendants of $i$. From a Markov chain analysis we then obtain the total rate of requests leaving the node (which equals the expected load of the edge incident on it in the direction of the root). In the next subsections we shall illustrate the Markov chain analysis.

## 4.2 Markov chain analysis for a single node

In this subsection we analyze the cache at a single node. From now on, we suppress the index $i$. Thus requests for files arrive with rate $\lambda$ and the probability that some request is for file $j$ is $p_j$, independent of all other requests. After a new request of a file, which is not in the cache, arrives at a node, the file is placed in the cache and, simultaneously, the least recently used file is removed from the cache. The state of the cache can be described by the files contained in the cache and the order in which they have recently been used; thus the total number of different states is equal to $\binom{N}{M}M! = N(N-1)\dots(N-M+1)$. Clearly, this process is a discrete-time Markov chain. The stationary probability that file $j$ is present in cache is denoted by $p_j^{(M)}$.

A closed form expression for the limiting distribution of the Markov chain can be obtained by using the fact that the LRU strategy for caching is closely related to the "move-to-the-front rule", which has extensively been studied in the context of selecting records from a computer file and taking out books from a library shelf. For example, in [5], Hendricks considers the following problem. A library contains $N$ different books $B_1, \dots, B_N$ arranged on a single shelf, and regardless of the arrangement of the books the probability of selecting $B_j$ is $p_j$, where $p_j$ $(1 \le j \le N)$ are positive numbers such that $\sum_{j=1}^{N} p_j = 1$. Only one book is demanded each time and the book is returned as the next book is borrowed. Upon return, a book is placed at the end of the shelf nearest to the librarian's desk. What is the stationary distribution of the order of books on the shelf?

Let $\tau = (j_1, \dots, j_N)$ be a permutation of $\{1, \dots, N\}$. The Markov chain has $N!$ states, corresponding to the $N!$ different permutations of the books. The state $B(\tau)$ corresponds to the books being arranged in the order, from left to right, $B_{j_1}, \dots, B_{j_N}$ on the shelf. Let $u$ be the equilibrium probability of state $B(\tau)$. Hendricks proved that

$$u = \prod_{n=1}^{N} (p_{j_n} / \sum_{r=n}^{N} p_{j_r}).$$

The shelf corresponds to the cache in our case, while books correspond to files. Hendricks' result can be applied if we only take into consideration which books are in the first $M$ positions and how they are ordered. Let $\pi = (j_1, \dots, j_M)$ be an ordered $M$-tuple, whose entries are distinct members of the set $\{1, \dots, N\}$, and let $v$ be the equilibrium probability of the corresponding state of our Markov chain. Then

$$v = \sum_{\sigma} \prod_{n=1}^{N} (p_{j_n} / \sum_{r=n}^{N} p_{j_r}),$$

where the summation is over all permutations $\sigma = (j_{M+1}, \dots, j_N)$ of the set $\{1, \dots, N\} \backslash \{j_1, \dots, j_M\}$.

Unfortunately, the above closed-form expression is not likely to be useful in practice, since calculating its value involves a summation of $(N-M)!$ terms. Typically $N - M = \Omega(N)$ (that is, $(N-M)/N$ is bounded below by a positive constant as $N, M \to \infty$), and thus $(N-M)!$ grows exponentially in $N$ and $M$. This implies that evaluating the formula quickly becomes computationally infeasible. Therefore, we propose the following approximation for $p_j^{(M)}$ :

$$p_j^{(M)} = 1 - (1 - p_j)^z, \tag{1}$$

where $z$ solves the equation

$$M = N - \sum_{j=1}^{N} (1 - p_j)^z. \tag{2}$$

Equation (2), which determines $z$, follows from summing equations (1) over all $j$ and using the fact that $\sum_j p_j^{(M)} = M$, as in equilibrium the cache should be full. Furthermore, the number $z$ represents the expected number of steps required to pick $M$ distinct files and hence $1 - (1 - p_j)^z$ can be interpreted as the probability that file $j$ has been selected during the $z$ steps required to form the current contents of the cache.

### 4.3 Expected loads on the edges of the tree

Finally, we show how to calculate the expected loads on all the edges of the tree. For each node $i$ and each file $j$, let $\bar{p}_{i,j} = \bar{\lambda}_{i,j}/\bar{\lambda}_i$ be the probability that a given request at node $i$ (either external or from one of the descendants of node $i$) is for file $j$. Clearly, if node $i$ is a leaf of the tree then $\bar{p}_{i,j} = p_{i,j}$ for $j = 1, \dots, N$. For each leaf $i$, we calculate $p_{i,j}^{(M)} = 1 - (1 - \bar{p}_{i,j})^{z_i}$, where $z_i$ solves $M_i = N - \sum_{j=1}^{N}(1 - \bar{p}_{i,j})^{z_i}$. The rate of requests leaving leaf $i$ equals $\lambda'_i = \bar{\lambda}_i \sum_{j=1}^{N} \bar{p}_{i,j}(1 - p_{i,j}^{(M)})$, and hence the rate of request arrivals for a node $i$ at height 1 from the bottom is $\lambda_i + \sum_r \lambda'_r$, where the sum is over all nodes $r$ that are descendants of node $i$. We proceed recursively in the manner described above to higher levels of the tree, until we reach the root. The rate of requests leaving a node equals the expected load on the edge leaving the node towards the root.

# 5 Conclusion

In this paper we have considered the problem of placing proxies in a network to get a better performance of the net. We have divided this problem into two subproblems: identify nodes where proxies will be placed and determine the size of the proxies. To make the problems easier to handle, first we have simplified the problems by neglecting the stochastic structure of the process resulting from the caching strategies used in practice. Based on the assumption that fixed files are placed in the proxies, we have developed algorithms to determine locations for and sizes of the proxies.

Since the estimated quality of the resulting proxy placement is calculated using the deterministic model, it may be not very realistic. To overcome this, in a second step we have presented a stochastic analysis for the commonly used LRU caching strategy to achieve a more realistic estimate of the quality of the solutions. This analysis may be combined with the deterministic algorithms in an iterative procedure: First, on the base of given bounds on the loads of the edges, calculate a solution which in the deterministic model achieves these bounds on the loads. Afterwards, analyze the solution using the stochastic method. Based on the outcome of this analysis, change the used bounds on the loads and iterate the procedure.

# References

[1] C. Aggarwal, J.L. Wolf and P.S. Yu, Caching on the World Wide Web, IEEE Transactions on Knowledge and Data Engineering 11 (1999), 94 - 107.

[2] H. Braun and K.C. Claffy, Web traffic characterization: an assessment of the impact of caching documents from NCSA's web server, Computer Networks and ISDN Systems 28 (1995), 37 - 51.

[3] R. Cáceres, F. Douglis, A. Feldmann, G. Glass and M. Rabinovich, Web proxy caching: the devil is in the details, Performance Evaluation Review 26 (1998), 11 - 15.

[4] S.J. Caughey, D.B. Ingham and M.C. Little, Flexible open caching for the Web, Computer Networks and ISDN Systems 29 (1997), 1007 - 1017.

[5] W.J. Hendricks, The stationary distribution of an interesting Markov chain, Journal of Applied Probability 9 (1972), 231 - 233.

[6] A. Luotonen and K. Altis, World wide web proxies, Computer Networks and ISDN Systems 27 (1994), 147 - 154.

[7] C. Maltzahn, K.J. Richardson and D. Grunwald, Performance issues of enterprise level web proxies, Performance Evaluation Review 25 (1997), 13.

[8] E. Markatos, Main memory caching of web documents, Computer Networks and ISDN Systems 28 (1996), 893 - 905.

[9] J. Shim, P. Scheuermann and R. Vingralek, Proxy cache algorithms: design, implementation and performance, IEEE Transactions on Knowledge and Data Engineering 11 (1999), 549 - 562.

[10] J. Wang, A survey of web caching schemes for the internet, ACM Computer Communication Review 29 (1999), 36 - 46.

[11] C.E. Wills and M. Mikhailov, Towards a better understanding of Web resources and server responses for improved caching, Computer Networks 31 (1999), 1231 - 1243.

# Displacement of a viscoplastic fluid in an inclined slot

B.W. van de Fliert and J.B. van den Berg

**Abstract**

The steady displacement of one viscoplastic fluid by another is studied in an inclined channel. The aim is a prediction of the finger width from simple balance laws. It is argued that no accurate prediction can be acquired from the far field velocity profiles only, but that instead a calculation of the two-dimensional behaviour near the free interface is needed. It is suggested that the static residual thickness can be determined from a minimization procedure for the dissipation in the system.

**Keywords**

Viscoplastic, Bingham fluids, Inclined channel, Drilling of oil wells

## 1 Introduction

The primary cementing operation for the drilling of oil wells consists of the following stages: first drill a new part of the well, trip-out the drill pipe, trip-in the steel casing, pump spacer or lead and tail slurry to displace the drilling mud upwards in an annulus and start again. For the study of the last stage, the miscible displacements of viscoplastic fluids for the purpose of well cementing, a simplified model is discussed of a two-dimensional channel with two Bingham fluids.

Known models for such a displacement of two fluids are based on lubrication type approximations and eccentric annular Hele-Shaw cells, see the references in [2]. During the Study Group attempts have been made (although unsuccessfully) to balance the effect of gravity, say the load of the displacer fluid $c$, to the lubrication pressure in the thin residual layers, comparable to a lift, using lubrication arguments. In the literature several lubrication scalings have been studied, which indicate that a valid approximation can be made of the velocity field, but not of the yield surfaces. We will not present a discussion of the usefulness of the lubrication models but only refer to the citations in [1],[2].

It seems important to investigate what happens on the gap scale, since numerical and analytical studies indicate that while the front of the displacer fluid moves steadily down the slot, a uniform layer of residual is left behind at the walls. The formation of these static residual layers is observed when the yield stress of the displaced fluid is not exceeded at the walls of the channel. The aim of this study is to predict the thickness of the static residual layers, preferably from a simple criterion and in an inclined channel. In the vertical, symmetric case an accurate criterion for the layer thickness seems to be given by a recirculation criterion. The first aim is therefore to understand the predictive value of the recirculation criterion and to generalize the idea to the non-symmetric case. It is shown in this report why preventing recirculation in the moving frame of reference gives a lower bound for the layer thickness. However, in the case of buoyancy, direct numerical simulations indicate that the recirculation criterion is not accurate and further investigations are required for the flow behaviour near the tip of the interface between the two fluids.

We start in section 2 with a reprise of the rheological properties of the Bingham fluids, the introduction of the dimensionless numbers, and the steady velocity profiles in a channel. In sections 3 and 4 we discuss the recirculation criterion for a vertical and an inclined channel. When a steady numerical calculation in a moving frame of reference is to be done, instead of solving the dynamic system, the difficulty arises of the non-uniqueness of the steady interface between the two fluids (the displacement front). The selection of the finger width and the residual layer thicknesses is likely to be found from a

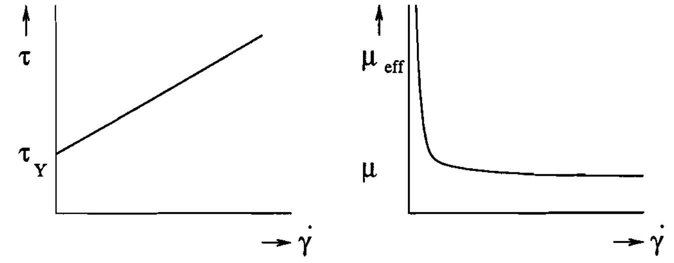

Figure 1: Bingham stress-strain plot and effective viscosity ($\mu + \frac{\tau_Y}{\dot\gamma}$).

minimization criterion for the dissipation of energy. In section 5 we discuss the numerical approach using a regularization of both the interface and the rheological properties.

## 2 Viscoplastic fluids

We will adopt the following notation for the viscoplastic Bingham fluids in a two-dimensional channel.

Suppose fluid $m$ (mud) is being displaced and fluid $c$ (cement slurry) is the displacer. Assuming that both fluids are perfectly Bingham, but with different rheological parameters, the constitutive laws with yield stress $\tau_Y$ and viscosity $\mu$ (with different values for fluids $m$ and $c$, in spatial domains $\Omega_m$ and $\Omega_c$ respectively) are given by:

$$\begin{array}{lcl} \dot\gamma(\mathbf{u}) = 0 & \Longleftrightarrow & \tau(\mathbf{u}) \le \tau_Y \\ \tau_{ij}(\mathbf{u}) = \left(\mu + \frac{\tau_Y}{\dot\gamma(\mathbf{u})}\right)\dot\gamma_{ij}(\mathbf{u}) & \Longleftrightarrow & \tau(\mathbf{u}) > \tau_Y \end{array} \tag{1}$$

where we have used the notation

$$\dot\gamma_{ij} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}, \; \dot\gamma(\mathbf{u}) = \left(\frac{1}{2}\sum_{i,j}\dot\gamma_{ij}^2(\mathbf{u})\right)^{1/2}, \; \tau(\mathbf{u}) = \left(\frac{1}{2}\sum_{i,j}\tau_{ij}(\mathbf{u})\right)^{1/2}.$$

To avoid viscous fingering the viscosity $\mu_c$ of the displacer $c$ is taken to be smaller than $\mu_m$ of fluid $m$; the static layers will typically occur when the yield stress $\tau_{c,Y}$ of fluid $c$ is smaller than that of $m$.

The equations of motion are made dimensionless relative to the mean displacement velocity $U_0$, the density of the displaced fluid $\rho_m$, and the slot half-width $D$, which gives as dimensionless numbers: the density ratio

$$r = \frac{\rho_c}{\rho_m} \ge 1,$$

the buoyancy parameter

$$b = \frac{(\rho_c - \rho_m)\,gD}{\rho_m U_0^2} \ge 0,$$

and the plastic yield stresses and viscosities

$$\tau_Y = \frac{\tilde\tau_Y}{\rho_m U_0^2}, \; \mu = \frac{\tilde\mu}{\rho_m U_0 D}.$$

(These should be read with $\tau_{m,Y}$ and $\tau_{c,Y}$, respectively $\mu_m$ and $\mu_c$.)

Typically there will be an interface between the two fluids, as indicated in figure 2, that moves steadily along the channel with some speed $S$. We change to a moving frame of reference in which the

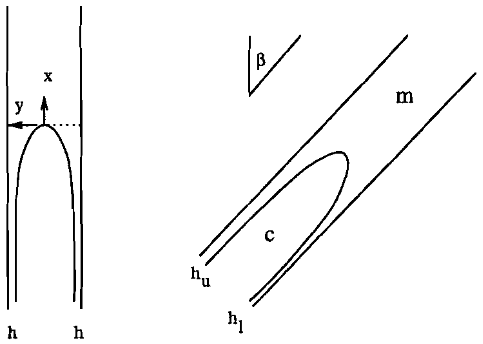

Figure 2: Steady interface between displacer $c$ and displaced fluid $m$; symmetric and non-symmetric front.

shape of the interface is fixed. We choose coordinates $(x, y)$ with corresponding velocities $(u, v)$, with $x$ in axial direction, the $y$-axis fixed by the tip of the displacement front, and with $\beta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ denoting the angle between the $x$-axis and the vertical, gravitational, direction. The channel thus gives a domain $\Omega = (-L, L) \times (-1, 1)$, where the dimensionless length $L$ is large (but finite), with $\Omega$ divided into fluid domains $\Omega_c$ and $\Omega_m$ separated by an interface $\Gamma$. At one end of the channel ($x = L$) there is only the fluid $m$, and at the other end ($x = -L$) there will be two static layers of fluid $m$ at the walls of the channel, say with thickness $h_u$ and $h_l$, so that fluid $c$ basically flows though a channel of width $(2 - h_u - h_l)$.

Conservation of volume gives the propagation speed of the front and therefore the proper speed for a moving frame of reference $S$,

$$S = \frac{2}{2 - (h_u + h_l)}. \tag{2}$$

The ratio of the (dimensionless) yield stress and the viscosity times the squared mean velocity is called the Bingham number of the fluid, so for fluid $m$, $B_m = \frac{\tau_{m,Y}}{\mu_m}$, and for fluid $c$, $B_c = \frac{\tau_{c,Y}}{\mu_c S^2}$.

In the moving frame of reference, the continuity and momentum equations read

$$\nabla \cdot \mathbf{u} = 0, \tag{3}$$

in $\Omega_c$ :

$$r\,\mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \tau - \mathbf{b}, \tag{4}$$

with $\mathbf{b} = b(\cos\beta, \sin\beta)$, and in $\Omega_m$ :

$$\mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \tau. \tag{5}$$

Boundary conditions are given by

$$u(x, \pm 1) = -S,\ v(x, \pm 1) = 0,$$

and at the interface $\Gamma$ the conditions are given by continuity of the velocities and continuity of the normal stresses.

## 2.1 Far field velocity profiles

The steady profile downstream, where only fluid $m$ is found, is a plane Poiseuille flow of a Bingham fluid; in a moving frame of reference with propagation speed $S$ this is given by

$$u(L, y) = u_L(y) = \begin{cases} \frac{3}{Y_m+2} - S, & |y| \in [0, Y_m) \\ \frac{3}{Y_m+2}\left(1 - \frac{(|y|-Y_m)^2}{(1-Y_m)^2}\right) - S, & |y| \in [Y_m, 1] \end{cases} \tag{6}$$

where $Y_m = \frac{1}{\xi(B_m)}$, $B_m = \frac{\tau_{m,Y}}{\mu_m}$ and $\xi(B)$ is the only root of the parametric cubic equation

$$2\xi^3 - \left(3 + \frac{6}{B}\right)\xi^2 + 1 = 0 \tag{7}$$

satisfying $\xi(B) > 1$.

Upstream there is, besides the static layers of fluid $m$ at the walls, a plane Poiseuille flow for fluid $c$,

$$u(-L,y) = u_{-L}(y) = \begin{cases} \frac{3}{Y_c+2Y_i} - S, & |y| \in [0, Y_c) \\ \frac{3}{Y_c+2Y_i}\left(1 - \frac{(|y|-Y_c)^2}{(Y_i-Y_c)^2}\right) - S, & |y| \in [Y_c, Y_i) \\ -S, & |y| \in [Y_i, 1] \end{cases} \tag{8}$$

where $Y_c = \frac{Y_i}{\xi(B_c)}$, with $\xi$ as defined in (7) for $B_c = \frac{\tau_{c,Y}}{\mu_c S^2}$. The position of the interface, at $Y_i = 1 - h$, is assumed to be symmetric in (8); without this assumption, we can simply shift the velocity profile $u_{-L}(y)$ over a distance $y_c = \frac{h_l - h_u}{2}$, to obtain two different interface positions at $y = Y_u$ $(= 1 - h_u)$ and at $y = Y_l$ $(= 1 - h_l)$. We remark that due to the scaling, the far field velocity profiles are independent of the buoyancy $b$ (even though the stress and the front speed $S$ are not). This implies that the position of the finger of displacer fluid $c$ (in the $y$-direction) is in a way independent of the buoyancy, or in other words, the thickness of the residual layers of fluid $m$ are to be determined solely by the dynamics around the tip of the finger.

For the existence of the static residual layers, it is necessary that the shear stress at the wall of the channel does not exceed the yield stress of fluid $m$. For example for the symmetric case, this translates in a maximal value of the thickness, $h_{\max}$, determined by the condition

$$\tau_{wall} = \frac{\tau_{c,Y}\xi(B_c)}{Y_i} + b(1 - Y_i) \leq \tau_{m,Y}. \tag{9}$$

This means that $h_{\max}$ is defined by $1 - \frac{1}{S_{\max}}$ with $S_{\max}$ the velocity at which equality is attained in (9), with both $B_c$ and $Y_i$ dependent on $S$. We see from this that the condition $\tau_{c,Y} > \tau_{m,Y}$ is indeed a necessary condition to find a $h_{\max} > 0$.

# 3 Recirculation criterion

In this section we discuss the symmetric case, in a vertical channel.

Apart from the maximal layer thickness $h_{\max}$, as determined by (9), another limit for the static layer thickness can be found by looking at the velocities at the centerline at $y = 0$. Since the mean velocity upstream at $x = -L$ is given by the speed $S$, or actually by 0 in the moving frame, the maximum speed at $y = 0$ has to exceed this, so $u_{-L}(0) > 0$, and the velocity has to drop down to zero when we are moving along the centerline towards the tip of the finger. Since the displacer fluid is pushing the fluid $m$ out of the channel, it can be expected that the maximum velocity will decrease further, so that $u_{-L}(0) > 0 \geq u_L(0)$. Here pushing is seen as a compressive stress, with $u_x \leq 0$, while an increase in velocity, $u_x \geq 0$ would imply that the fluid $m$ is pulling the displacer $c$ out of the channel. The inequality $0 \geq u_L(0)$ predicts that there will be no recirculation downstream (in the moving frame) and translates into a critical value of the layer thickness. This value for $h$ at which $0 = u_L(0)$ is denoted by $h_{circ}$ and it gives a lower bound for the speed $S$; from (6),

$$S \geq \frac{3}{1/\xi(B_m) + 2} \left(= \frac{1}{1 - h_{circ}} = S_{circ}\right).$$

It is observed in numerical calculations of the displacement in a vertical channel, that $h_{circ}$ is in fact a rather good predictor for the value of the actual thickness $h$, i.e. equality is nearly attained. How can this be understood? Consider the situation when there is only a small compressive stress in fluid $m$, which means that at the centerline $y = 0$ the total stress $|\tau|$ is certainly smaller than $\tau_{m,Y}$. Along the

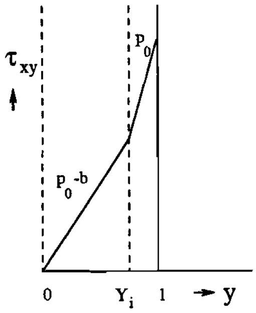

Figure 3: Shear stress in the two Bingham fluids, upstream at $x = -L$.

centerline, where also the shear stress vanishes, this results in a rigid movement and the velocity has to be equal to the velocity at the interface. It can thus be expected that when the yield stress of $m$ is sufficiently large, the recirculation criterion is very accurate. However, if the yield stress $\tau_{m,Y}$ is close to that of fluid $c$, it should be expected that the speed $S$ is larger than the critical recirculation speed. Indeed, since the interface causes a fully two-dimensional flow, there have to be compressive stresses $\tau_{xx}$ and $\tau_{yy}$ in fluid $c$, that exceed the yield stress $\tau_{c,Y}$ (since $\dot{\gamma} \neq 0$ but $\tau_{xy} = 0$ along $y = 0$) and since the normal stresses are continuous over the interface, they will exceed the yield stress of $m$ and the velocity decreases further along the centerline. Similarly, if the viscosity of $m$ is relatively large, the stresses near the tip will be big enough to "melt" the fluid $m$ at the centerline, and the speed deviates from the recirculation speed. These effects are not all that clear in the data in [2] where comparisons of $h$ and $h_{circ}$ are shown using variations in the rheological parameters (figures 15 and 16 in [2]). The argument does give an idea, however, why the layer thickness decreases (with a decrease in $S$) with increasing yield stress or decreasing viscosity of fluid $m$, as is observed in the numerical computations.

We remark that the value of $h_{circ}$ is fully determined by the far field conditions, which are independent of the buoyancy $b$. When looking at the data of the layer thickness for different values of the buoyancy $b$, see figure 4 below for $\beta = 0$, we observe that the actual layer thickness is larger than predicted by the recirculation criterion (as can be expected from the argument above), but furthermore that it varies with $b$.

We expect that for larger values of $b$ the recirculation criterion becomes more accurate (see also figure 4 for $\beta = 0$). In figure 3 a plot of the shear stress is shown in the far field with residual layers. The slope of the stress in the center part is given by the modified pressure $\tilde{p}_0 = p_0 - b = \tau_{c,Y} S \xi(B_c)$, where $p_0$ denotes the pressure gradient in the channel that is applied to achieve a throughput of 2 (after scaling). This means that in the static layers of fluid $m$, the slope of the shear stress is given by $\tilde{p}_0 + b$. The maximal layer thickness $h_{\max}$ is determined by the shear stress at the wall, as given in (9). From the figure we can conclude that when $b$ is increased substantially, $\tau_{wall}$ will increase and $h$ will need to decrease, since a static layer can only exist if $h$ does not exceed $h_{\max}$; but there is a lower bound for $h$ given by $h_{circ}$. It is therefore expected that for large $b$, the front speed $S$ will approach $S_{circ}$. This is related to the idea that if the material that is pushed away is very light ($b$ large), then it is simply pushed away without high stresses, which means that the yield stress is not exceeded at the centerline in the light fluid, therefore, by the argument given above, $S = S_{circ}$.

# 4 Inclined channel

When we consider the case of an inclined channel at angle $\beta$, with buoyancy parameter $b$, the same argument as before can be given for a lower bound of the front speed $S$. Suppose that the tip of the finger (where the speed vanishes exactly) lies within the $y$-interval $[-Y_m, Y_m]$, where the fluid $m$ downstream behaves like a solid, then $S \geq S_{circ}$ as before. Note that at any point on the interface away from the tip, the velocity $u$ will be smaller, so that when the tip does not lie within the solid-interval, the estimate will be less sharp.

It should be noted again that the recirculation criterion contains neither buoyancy or inclination angle and indeed, the criterion does not give an accurate correspondence with numerical results in an inclined channel. We use the numerical data provided in [2]. Some simple data fitting indicates that the plots may be linear in $\sin\beta$ and $\cos\beta$, but they are not linear in the buoyancy parameter $b$. The data also seem to indicate that both the total finger width $2-(h_l+h_u)$ and the position of the centerline of fluid $c$, $(h_l-h_u)/2$, are monotonic in the buoyancy $b$ and the inclination angle $\beta$.

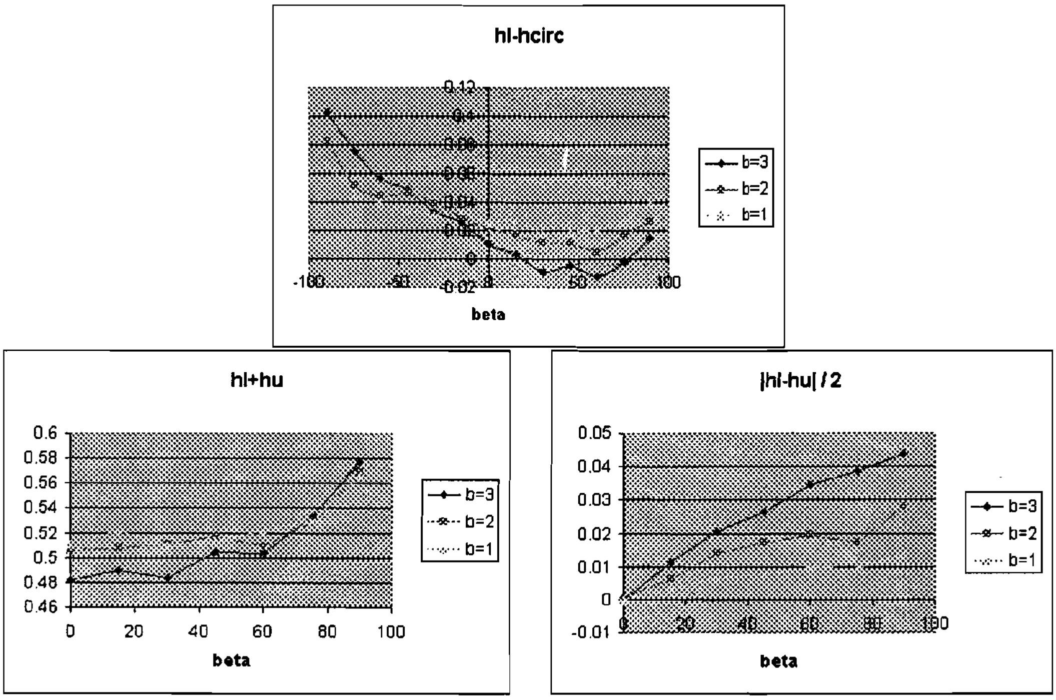

Figure 4: Thickness of the residual layer $(h_l - h_{circ})$ for inclination angle $\beta$; the lower two figures show the total layer thickness and the centerline position $y_c$.

Since the layer thicknesses, upper and lower, depend on the buoyancy and inclination angle, while the volume conservation (2) and recirculation criterion do not, additional information should be gained from the momentum equation. Unfortunately, the momentum equation cannot be integrated over the far field profiles only, since most of the viscous dissipation occurs near the front of the interface. The incompressibility of the two-dimensional material allows one to visualize the flow using the contourlines of the Stokes streamfunction, defined by $u=\psi_y, v=-\psi_x$. The fact that there is no recirculation in the moving frame, implies that the contourlines of the streamfunction do not have large gradients, which can be interpreted as a minimization of viscous dissipation.

If the inner product is taken of velocity $\mathbf{u}$ and (4) and (5), and integrated over $\Omega$, ignoring inertia effect for simplicity, we observe

$$\oint_{\partial\Omega} -p\mathbf{u}\cdot\mathbf{n} + \int_{\Omega} \tau : \nabla\mathbf{u} + \int_{\Omega_c} bv\sin\beta = 0. \tag{10}$$

Here we have used that over the interface, the pressure, the normal stress and the velocities are continuous, so that contributions from the two fluids balance:

$$\oint_{\Gamma} -p\mathbf{u}\cdot\mathbf{n} = \int_{\Gamma} (\tau\cdot\mathbf{n})\cdot\mathbf{u} = 0,$$

while at the outer boundaries the no-slip condition gives

$$\int_{\partial\Omega} (\tau \cdot \mathbf{n}) \cdot \mathbf{u} = 0,$$

and, finally, using that the interface $\Gamma$ is a streamline, which can been defined as the contour for $\psi = 0$, we find that

$$\int_{\Omega_c} bu \cos\beta = \int_{-L}^{0} b\psi \cos\beta]_{y=-Y_i(x)}^{y=Y_i(x)} \, dx = 0.$$

Again using the no-slip condition at the channel walls, the first integral in (10) reduces to contributions from the far field, at $x = -L$ and $x = L$, where the pressure and velocity are known explicitly, given by the Poiseuille flow in (6) and (8). Writing for the pressure gradient along the channel $p_0$, we find that near $x = -L$,

$$p(x, y) = -p_0 x - b \sin\beta(y - 1 + h_u),$$

while towards $x = L$ simply

$$p(x, y) = -p_0 x.$$

This means that the first integral in (10) is given by $4p_0 L + 2b \sin\beta(1 - \frac{h_l - h_u}{2})$. The third integral can also be calculated using the expression for the pressure, using that

$$\int_{\Omega_c} bv \sin\beta = -\int_{-Y_i}^{Y_i} b\psi \sin\beta]_{x=-L}^{x=Y_i(y)} \, dy = \int_{-Y_i}^{Y_i} pyu \sin\beta \, dy = 2b \sin\beta \, \frac{h_l - h_u}{2}.$$

We thus conclude that the viscous dissipation is given by

$$\int_{\Omega} \tau : \nabla \mathbf{u} = -2\left(2p_0 L + b \sin\beta(1 - h_u)\right), \tag{11}$$

where the last term can be written using $S$ in (2). Observe that this last term does not depend on $2L$, the length of the channel under consideration, therefore this dissipation takes place in a localised region near the tip of the finger. This dissipation can not be determined without explicit knowledge of the shape of the free interface and the corresponding flow near this interface and therefore it cannot be expected to supply a simple integral criterion for the flow characteristics. Both the interface and the stresses near this interface need to be calculated in a dynamical simulation or other criteria have to be found from a numerical calculation of the steady case.

Dynamical calculations have been done by Schlumberger previously [2], based on the dynamic problem with fully two-dimensional displacement computations in a volume-of-fluid-method. Most of the computations were done with rheological parameters: $(b, \tau_{c,Y}, \tau_{m,Y}, \mu_c, \mu_m) = (0, 0.2, 0.5, 0.01, 0.05)$, and $(0, 0.2, 1.0, 0.005, 0.01)$, for which $h_{circ} = 0.13$ and $0.04$ respectively. The data shown in figure 4 below, however, are from calculations in an inclined channel with $h_{circ} = 0.23$. Steady calculations have not been done, but we would like to make some remarks about the difficulties with such computations.

# 5 Numerical approach

It is shown in [1] that there does not exist a unique solution for the steady interface, in the sense that given an interface, sufficiently smooth, a solution of the velocities or streamfunction can be found in the two fluid domains $\Omega_c$ and $\Omega_m$. This implies that the selection of the interface is a dynamic effect or that it is selected by a minimization principle, for instance by minimization of viscous dissipation. The recirculation criterion and the contourplot of the streamfunction seem to indicate that the viscous dissipation is a good selection criterion, but numerical calculations should substantiate or gainsay such a claim.

The numerical approach is to do a regularized problem, without a sharp interface and by smoothing out the rheological properties. The interface regularization can be done by modelling the displacement as the advection of a passive scalar, say a concentration $C$; this is achieved by replacing the kinematic condition at the interface $y = Y_i(x, t)$ by the advection

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} + v\frac{\partial C}{\partial y} = 0$$

with $C = 1$ in $\Omega_c$ and $C = 0$ in $\Omega_m$. This gives a diffuse region of intermediate concentration instead of a sharp interface. The intermediate concentration values are only observed in a thin region determining the smoothened interface; for any meshpoint in this region the concentration-dependent rheology can be used, for example $\mu(C) = C\mu_c + (1 - C)\mu_m$.

The effect of the smooth interface is that the displacement is considered as a flow of only one fluid, with different rheological properties at different positions. Another way to dismiss the interface by considering the fluids to be effectively one fluid, is to fix the rheological parameters as a function of position determined by the zero contourline of the previous iterate for the streamfunction $\psi$.

The effective viscosity of this fluid is regularized by using

$$\mu_{eff} = \mu + \frac{\tau_Y}{\dot{\gamma}(\mathbf{u}) + \varepsilon}$$

with $\varepsilon$ a fixed, small parameter, and all the other parameters depending on the concentration $C$ or the position.

In a streamfunction formulation, the steady problem is now reduced to a fourth order problem in $\psi$, with only boundary conditions to be provided at the boundary of the specified domain $\Omega$ (so without interface). The conditions at the channels walls are the no-slip conditions, $u = -S, v = 0$, and at the in- and outlet the conditions are dictated by the far field velocity profiles. The only unknowns in these conditions are the layer thicknesses $h_u$ and $h_l$, which will be the parameters over which the dissipation is minimized, for example in a steepest descent approach. In the streamfunction formulation, the thicknesses are determined by the distance of the zero contourline from the wall; in the case where the interface is regularised with a concentration $C$, the position of the interface should be found from interpolation. We do not expect these steady calculations to be more cost efficient than the dynamic problem, but it may provide insight in the selection of the front and the corresponding layers.

## Acknowledgement

The problem was brought to the study group by M. Allouche and I.A. Frigaard, Schlumberger Dowell, Etudes & Productions, Clamart, France. Contributions in study group discussions by M. Bowen, H. Margaretha, T.J.H. van Rossum, J.H. Westhuis.

# References

[1] I.A. Frigaard, O. Scherzer, G. Sona. Uniqueness and Non-uniqueness in the Steady Displacement of Two Visco-plastic Fluids. Preprint, submitted to *ZAMM,* 1999.

[2] M. Allouche, I.A. Frigaard, G. Sona. Static Wall Layers in the Displacement if Two Visco-plastic Fluids in a Plane Channel. Preprint, submitted to *J. Fluid Mech.,*1999.

# Route information from a central route planner

G. Hek, G. Lunter, J.A.M. Schreuder, J. White

**Abstract**

We present a discussion of a problem posed by researchers of the company Ericsson, namely, to estimate the fraction of the road users in a road network that must participate in a central route planning scheme such that travel time predictions improve significantly. A road user who participates is expected to inform the central route planner of his intentions to travel from an origin to a destination and is expected to travel along the route advised by the planner.

The aim of this work is to derive a measure of travel time performance depending on the number of road users who are participating in the central route planner. The approach is mainly of a statistical nature.

**Keywords**

Central route planner, Traffic flow estimation, Traffic flow control, Nash equilibrium.

## 1 Introduction

Ericsson is interested in developing a "central route planner". The function of a central route planner (CRP) is to advise road drivers on journey routes. Specifically, before travelling from one location to another, a driver uses the telephone to query the central route planner, which tells the driver the fastest route to take, an estimated journey time and possibly other information such as reliability estimates or worst-case scenarios. Ericsson must decide how the central route planner will calculate the routes and times it distributes, and which of the various available sources of data providing information on traffic flow they should use in making these calculations.

In particular, Ericsson would use historical data on traffic densities (possibly correlated with variables such as the day of the week, season, and weather forecast). However, Ericsson has also considered using the number of user queries themselves, for a given particular route, in addition to this historical data. The traffic forecasts made by the central route planner may be improved by doing this. Ericsson would like a measure of the "improvement" and wants to know how this "improvement" would depend on the percentage of drivers using their service. Their main question to us was: "How many user queries does Ericsson need to significantly improve upon the historical data predictions?"

At the moment Ericsson knows very little about this type of problem, and wants some advice on various issues. Perhaps not surprisingly, given the economic importance of efficient road networks and the current traffic jam problems, behaviour of traffic on road networks has been studied greatly. In section 2 we give a sketchy overview.

Ericsson's problem has many different aspects, some of which are discussed in section 4. Because of this scope, solving Ericsson's problem in full generality proved impossible. To get anywhere we had to make strong simplifications and restrictions, and ignore several important aspects of the problem. The simplifications we chose to make are discussed at the start of section 5, and the rest of that section is devoted to a statistical model of the relation between CRP users and travel time.

One part of this model is a relation between number of drivers on the road, and average travel time. In section 6 we use a simplified version of a traffic model we found in the literature to obtain a reasonable-looking approximating formula.

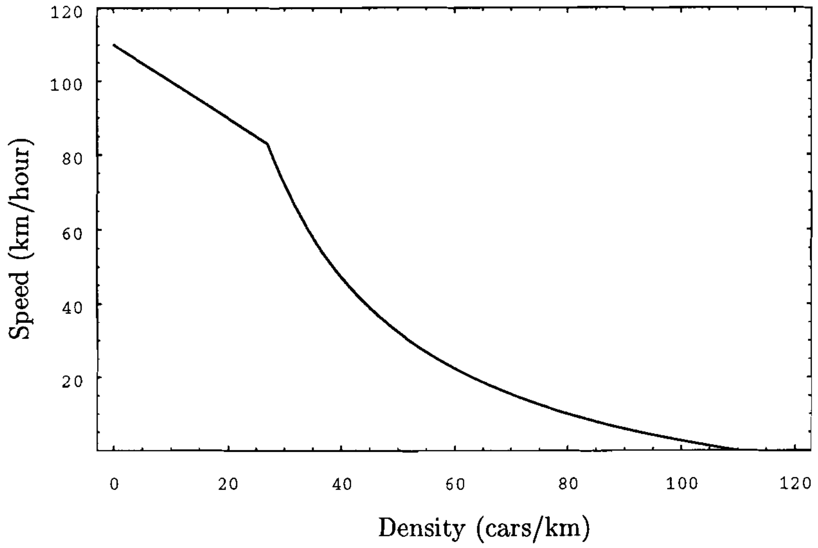

Figure 1: Smulders' function, relating density and equilibrium speed of traffic flow.

Section 7 gives concluding remarks. In our opinion, our analysis and the many different aspects of the problem that cannot be influenced indicate that a large number of users is necessary to significantly improve upon historical data predictions. We doubt whether such large numbers of drivers will cooperate in a central route planner system, and we think that Ericsson's idea will probably not be feasible.

## 2 Literature

The following is a selection of literature we used for this report. More references can be found in e.g. [Smu96] and [RB94].

In [Pay71] a traffic flow model is developed, inspired by fluid dynamics. The starting point is a set of partial differential equations. These equations are discretised, and terms are added to model the tendency of drivers to look ahead, and to adapt their speed to other traffic.

In [Smu96] this model is further developed. The notion of "equilibrium speed" is introduced, and a relation between traffic intensity (cars/km) and equilibrium speed is proposed. This speed, as a function of intensity, is continuous and monotonically decreasing, with a sudden steep drop at a critical density, marking the onset of traffic jams; see figure 1. This work also deals with dynamically influencing traffic flows. The analysis concerns a single highway segment. Influencing traffic flows is also studied in [dR94]. Methods for dynamic routing or dynamic traffic assignment, as opposed to static traffic assignment through signposts, are discussed.

A system theory approach to route planning, using both deterministic and stochastic models, can be found in [RB94]. The emphasis is more on the network, and less on modelling the traffic flow on single road segments, for which rather crude models are used.

Currently in the Netherlands, enough real-time information on traffic densities is available to make short-time density predictions feasible. A research group is using a model described in [vS99a] of traffic on the Amsterdam ring, based on [Smu96] and ideas from dynamic game theory, to predict densities a few hours in advance [vS99b]. These predictions may be used to drive an electronic messaging system.

In [Mie00] the idea of combining a toll-system (*rekeningrijden*) and advance booking was brought up. It contains elements of Ericsson's idea, and was investigated by the engineering consultants Niema, who concluded that the idea could be a "worthwhile contribution" to solving the traffic jam problem. Currently the idea is being discussed with several parties involved; see also [Nie00].

# 3 Some definitions

We define here a few terms that we shall use throughout. A *user* is a road user who has telephoned Ericsson's central route planner, and is following the advice provided. A *network* is a directed graph of road segments (in practice, each edge usually has a complementing one going in the other direction). We use the word *road* to denote a single edge in the network, that is, a segment without crossings, on- or off-ramps. For each edge or abstract road we define the following quantities, some of which are time-dependent:

- Length, $L$; corresponds to length of road.
- Number, $N$; corresponds to number of cars on road.
- Density, $D = N/L$; corresponds to number of cars per kilometre on a given road. For each road, we assume homogeneity.
- Intensity, $I$, also called 'traffic flow'; corresponds to number of cars driving past a certain point per hour.
- (Average) speed, $S$; average number of kilometres cars travel per hour on a given road.
- Capacity; maximum intensity, reached for some optimal density and speed.
- Travel time, $T = L/S$.
- Equilibrium speed; speed of traffic in stationary state, at a certain density.

Detailed models differentiate between equilibrium speed and the actual current (average) speed. In the models we use we shall not make this distinction.

By *network flow parameters*, we mean capacity, density, intensity and average speed for each road in the network. Capacity clearly depends on the *type of road*, i.e. number of lanes, highway or not, speed limits, etc. It also depends on *road conditions*, by which we mean all variables that influence the capacity of a road such as weather conditions, daylight, construction work, accidents on the road (or on the road going in the other direction, which may create a *kijkfile* (spectator jam). Finally, *routing information* is the information a user obtains when he[1] calls the central route planner. This information includes the estimated time his planned journey will take, the time at which he has to leave, and the reliability of the estimate.

# 4 Aspects of the problem

In this section we identify and comment on several aspects of the traffic estimation and routing problem. Many of these aspects have not found a way into the proposed model; however, we believe they are all relevant, and important to keep in mind when a decision about follow-up research on the CRP is to be made.

## 4.1 Two main approaches to the traffic problem

Vaguely put, the goal of traffic routing is to make more efficient use of available network capacity. Two main approaches may be identified (see also [RB94]), which we dub the *top-down* and *bottom-up* approach.

- Top-down: Guides traffic so that the total capacity of the network is maximized at a global optimum;
- Bottom-up: Provides users with accurate information and predictions, enabling them to choose the most efficient (fastest) route, i.e. every user is in a local optimum.

[1] Whilst not making strained efforts for political correctness, we are aware of the existence of female drivers.

The first is the approach taken by the government, when they try to influence traffic, for instance by imposing speed limits. The second approach is taken by individual drivers, when they choose departure time and route in order to encounter fewer jams.

Note that the two approaches use different notions of optimality. Government is interested in average throughput, an individual user is interested primarily in his own travel time. And indeed, these differing notions give rise to different optimal configurations. For example, within some bandwidth, imposing a speed limit results in a higher road capacity (the number of cars per minute flowing through), although individual cars take longer to arrive. Other even more counterintuitive situations may occur; see section 4.3.

## 4.2 Historical data and users information

We assume Ericsson has access to historical data on the network flow parameters. It is unclear to us whether this assumption is justified. We believe that a systematic, detailed and extensive database of past network usage is vital for predicting the traffic flow, and running a CRP service. Rijkswaterstaat routinely compares the traffic flow and speed of the flow with values of the recent past. Each traffic control center has a module for this.

The problem of predicting traffic flow using historical data alone is not trivial. A reasonable idea seems to identify independent, explaining variables, and use these to look up relevant past traffic situations. The independent variables would include at least the day of the week, the time of the day, the season and weather (forecasts). The database would provide an estimate of traffic intensity, as well as an estimate of road conditions, together with the resulting traffic densities.

Ericsson's extra source of information are the *user queries.* The information on users consists of the information about their current travel plans, and probably also of a database containing the travel histories of all users. This information may be used to improve the traffic intensity estimate. Using the database or a traffic model, this can subsequently be related to an improved traffic density estimate.

As the extra information contained in the user queries would almost certainly be related to network usage (intensity), rather than road conditions, it is important to compare how traffic flow varies with varying network usage, as compared to the variation due to varying road conditions. See section 4.5 for more remarks.

To determine the effect that using the extra information contained in user queries will have on the precision of estimated journey times, it is vital to look at the relationship between the number of drivers on a road, and the number of user queries (pertaining to this road). There may not in fact be a useful relationship between these quantities, for the following reasons.

Some proportion of the drivers may never call, for instance because they are bound to fixed departure times, or simply because of unfamiliarity with the system. People taking the same route regularly may not bother to call (often), especially if the proposed route and departure times do not vary much. An assumption on the relationship, for instance that a fixed percentage $\lambda$ of the drivers call Ericsson, will therefore probably not hold in general, but might however hold for the group of occasional drivers.

The CRP can provide best routing information after all user queries are collected. This would mean that users must call twice to obtain the requested information, which is not practical. Moreover, they must plan their journey well in advance, which may not be possible or desirable for everyone. Alternatively, the routing information may be continuously updated, with the system always giving the latest predictions. This has the disadvantage of rewarding late callers, reducing the effectiveness of the system.

Users may also give unreliable information, particularly if this information must be given well in advance, or ignore the CRP's advice, causing further problems. User input will hardly influence the information they get from the CRP, so that there is little incentive for the user to be very precise.

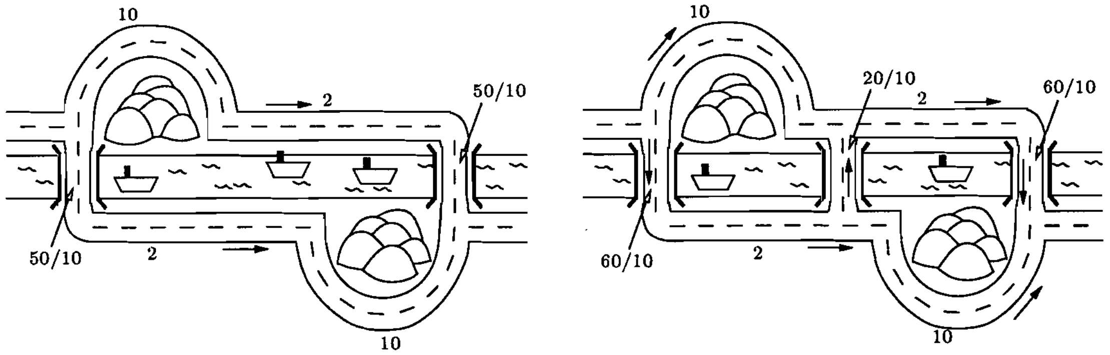

Figure 2: Braess' paradox. The numbers indicate the travel times for the corresponding edge; see text. Adding an edge, in this case a third bridge across the canal, leads to increased travel times for all drivers. Such paradoxical results have been observed in practice.

## 4.3 Nash equilibrium, Braess' paradox and the Prisoner's dilemma

One idea that came up during the discussions, is whether Ericsson, by advising their users properly, could "manage" traffic so that a more efficient use of the network would result, and hence better journey times for many of Ericsson's users. However, paradoxical situation may occur, as for instance noted by Frank Kelly [Kel91]:

> [I]f drivers are provided with extra information about random delays ahead, the outcome may well be a new equilibrium in which delays are increased for everyone.

In this section we shall make some remarks about a similar paradox, which states that adding edges to a traffic network may similarly increase delays.

It is usually assumed that drivers choose routes so as to optimize their own situation. Assuming full information, this leads to a traffic situation where every driver uses a *locally optimal* route, meaning that choosing a different route will not result in a decreased journey time for this particular driver. This is called a *Nash equilibrium*. More than one Nash equilibrium may exist.

Given that Nash equilibria are locally optimal, it is perhaps surprising that the *globally optimal*, or most efficient, traffic situation may *not* be a Nash equilibrium. For this statement to be meaningful, we need a definition of "efficiency". Here we choose as efficiency measure, any (weighted) average of the journey times experienced by all drivers in a given traffic situation. More precisely, the statement is as follows. For certain networks, traffic situations exist where, compared to the most efficient Nash equilibrium, *every* driver has decreased journey time. These traffic situations are more efficient according to our efficiency measure.

The fact that a Nash equilibrium need not be optimally efficient, is closely related to *Braess' paradox* [Bra68]. Braess found that adding an edge to a network may lead to a change in a Nash equilibrium, with *increased* delays for everyone, even though drivers have more routes to choose from. This phenomenon has actually been observed in practice [Tim90].

An example network where Braess' paradox occurs is given in figure 2; see also [Wai]. A network around two mountains and across a canal offers two alternative routes. Suppose that the delays on the various parts of the network are: 10 minutes to go around a mountain, 2 minutes to drive along the canal, and $n/10$ minutes to cross the canal, where $n$ is the number of cars using the bridge. This term models the congestion, due for instance to the narrowness of the bridge. Suppose 100 cars want to cross the canal. In this case the unique Nash equilibrium is reached when cars distribute themselves equally among the two available routes. The associated delay is $10 + 2 + \frac{50}{10} = 17$ minutes for each route.

Suppose now that a third bridge is constructed, as indicated in the right panel in figure 2. Two new routes are available, one going around both mountains, the other avoiding going around either. The first will not be used, but the second route is fast, with initially $\frac{50}{10}+2+0+2+\frac{50}{10} = 14$ minutes delay, so that the previous traffic situation is no longer a Nash equilibrium. The new unique Nash equilibrium is reached when 20 cars take the new route, with the remaining 80 divided equally among the two old routes. The delay in this situation is 18 minutes for all routes, longer than before.

The previous traffic situation, with 17 minutes delay, is still possible in the new network configuration. However, it is not a Nash equilibrium and will not spontaneously occur in practice: since the shorter route is faster, with only 14 minutes delay, taking this route is beneficial for the individual driver, even though it is detrimental to the "community".[2] This situation is analogous to the classical Prisoner's dilemma.

One way to induce drivers to move towards the non-Nash optimally efficient situation is to artificially increase the "cost" of the short route, for instance by imposing a monetary fine (essentially the content of the NIEMA proposal, see [Mie00, Nie00]). Drivers will weigh the benefit of a decreased delay against the monetary cost associated to the quicker route. A new equilibrium will set in, which is a Nash equilibrium associated to the weighted graph, with weights that include both the delay along the edge, and the monetary cost involved. With fines chosen appropriately, such a Nash equilibrium can correspond to an optimally efficient traffic situation.

We conclude that the two approaches, top-down or global, and bottom-up or local optimizing (see also section 4.1), result in different optimal solutions. Efforts for globally optimizing network usage are best done at the government level. Ericsson, on the other hand, is interested in the local problem of predicting travel time and the best possible route, in order to provide a service to users. For this reason, we abandoned the idea of global traffic management, and instead focused on predicting traffic densities, in order to find the fastest route for individual users.

## 4.4 Finding the shortest route

Once road speeds are estimated, finding the fastest route is relatively easy. It can be basically done with Dijkstra's shortest path algorithm (mind the datahandling!) for directed weighted graphs, adapted to take into account that the weights of the edges (the travel time along this edge) vary with time. Provided that the weights do not vary too quickly (as otherwise waiting before taking an edge may get you to your destination quicker than leaving immediately – Dijkstra's algorithm does not handle this correctly), this can be done easily. Such an algoritm has polynomial time complexity. For an application of Dijkstra's algorithm in a real-life situation see [Sch98].

## 4.5 The explaining variables of traffic jams

To provide reliable information, we want to predict how quickly a user can travel on the road network. This depends on the user (whether he is driving a truck or a car, whether he will drive fast or slow if he has an option), and on traffic conditions (the maximum attainable speed on each leg of his journey). Traffic conditions depend on several things, such as the type of road, the road conditions, the number of drivers on the road, the type of vehicles (cars or trucks), and previous traffic conditions (a jam takes time to dissolve; above criticality, a jam reinforces itself).

Let us focus on two main variables, demand and road conditions. Since the central route planner bases its estimate on an improved estimate of traffic density, and since Ericsson's additional source of information only carries further information on the first variable, demand, it is important to know something about the relative importance of these two variables for traffic density estimates.

A very simple first approach to answer this question empirically could be as follows. Instead of looking at the average speed, which varies over the roads of the network and may be difficult to measure, we look at the length of a traffic jam. This may be regarded as a stochastic variable. The lengths of traffic jams are already being measured and broadcast on the radio. Correlating these with variables related to demand and road conditions then gives information on how these variables explain the lengths of traffic jams. Real-time measurements of intensity are already done at some points in the Dutch road network (see [vS99a]).

[2] Choosing routes according to a globally efficient traffic situation is, in Hofstadter's language, following a "superrational" strategy (see [Hof85, Part VII]). His experiments indicated that rational people do not follow such strategies.

A potential problem in this approach is that the relationship between traffic jam lengths and demand is probably highly nonlinear: below a certain threshold, jams are very unlikely to occur. Because of this nonlinearity, ordinary linear correlation might not be the best way to measure the dependence between these variables.

# 5 A statistical approach

Here we present a model to answer Ericsson's question quantitatively, in a simplified setting.

## 5.1 Simplifying assumptions

All network flow parameters are important for e.g. simulations. ¿From the users' perspective, the average speed (or travel time) are mainly of interest. In order to predict the total travel time for a single user, Ericsson needs to be able to predict the travel time on each leg of the network, at every instant of time.

Once these estimates are known, finding the fastest route is relatively trivial. Therefore we shall focus on the precision of the journey time estimate, which depends on the "reliability" of network flow parameter estimation. Many of the variables that influence this are inherently stochastic in nature (such as the occurrence of accidents), so a statistical approach seems to be natural.

In order to simplify further, we will not consider the whole network or discuss the correlations in time mentioned, but focus on the problem of predicting the travel time on a *single* given road.

As final major simplification, we assume that the network flow parameters (such as intensity) are independent of time. In practice, this means that we shall consider a short time interval, where conditions can be regarded as being constant (but see section 6).

## 5.2 The model

In this model, we consider a single road. Let $N$ denote the number of cars on this road and $T$ the time taken to travel along the road. As mentioned previously, we assume there is a deterministic relation between the density and the speed, so that $T$ is some function of $N$:

$$T = g(N) \tag{1}$$

say. It is reasonable to assume $g$ is monotonically increasing and hence injective (see also section 6). Some of these $N$ drivers call Ericsson; say $U$ users. The problem is now to estimate $N$ given $U$.

First we need to model the distribution of $N$ itself. All we know of $N$ is that it is a discrete variable. We assume $N$ has a Poisson distribution with parameter $\mu$, say.

Furthermore, we assume that the probability of a driver being a user is $\lambda$. In other words, the conditional distribution of $U$ given $N = n$ is a binomial distribution with $n$ trials and parameter $\lambda$. The actual value of $\lambda$ may be deduced from historical (users) data. Then $U$ is Poisson distributed with parameter $\mu\lambda$.

We have historical data for a particular road, i.e. with frequencies $f_1, \dots, f_k$ there are $n_1, \dots, n_k$ cars on the road. The travel times are then $t_1, ..., t_k$ where $t_i = g(n_i)$.

In practice many $n_i$-values can occur, while the historical data may be limited. It is therefore probably a good idea to use *intervals* $[n_i, n_{i+1})$ of some appropriate length, instead of points. See also section 5.3.

Suppose we observe $U = u$ say, and we know

$$P(N = n_i | U = u) = \frac{P(U = u | N = n_i) P(N = n_i)}{P(U = u)}$$

by Bayes Theorem. Now $P(U = u | N = n_i)$ is given by the above assumption, $P(N = n_i) \simeq$

$f_i / \sum f_i$ by our assumption on the historical data, and

$$P(U = u) = \sum_{n_j \geq u} P(U = u|N = n_j)P(N = n_j).$$

As $n_i$ varies, this gives us the posterior distribution of $N$, given $U = u$. Hence, using that $g$ is injective (see also figure 5),

$$P(T = t_i|U = u) = P(g(N) = g(n_i)|U = u) = P(N = n_i|U = u) \tag{2}$$

gives the posterior distribution of $T$ given $U = u$. We finally estimate $T$ from a given $u$ by choosing the value $t_i$ which maximizes (2).

Let us now use our knowledge of the distributions of $N$ and $U$. We have

$$P(N = n) = \frac{e^{-\mu}\mu^n}{n!}, \quad P(U = u|N = n) = \binom{n}{u}\lambda^u(1-\lambda)^{n-u} \quad \text{and} \quad P(U = u) = \frac{e^{-\mu\lambda}(\mu\lambda)^u}{u!},$$

and a little algebra yields the posterior distribution of $N$:

$$P(N = n_i|U = u) = \frac{e^{-\mu(1-\lambda)}(\mu(1-\lambda))^{n_i - u}}{(n_i - u)!},$$

a Poisson distribution with parameter $\mu(1-\lambda)$, translated by $u$. It reaches its maximum at $N = u + \mu(1-\lambda)$, which is also its mean. Its variance is $\mu(1-\lambda)$. Now, $T = g(N)$, so that

$$\text{var}(T|U = u) \approx \text{var}(N|U = u)\left(g'(E(N|U = u))\right)^2 = \mu(1-\lambda)\left(g'(u + \mu(1-\lambda))\right)^2 \tag{3}$$

(Note that the quality of this approximation depends on the smoothness of $g$. In our case $g$ is nondifferentiable, see section 6, and (3) will be an underestimate just below critical densities.) An estimate of the variance of $T$ without user information is $\mu g'(\mu)^2$. With user information, this changes to (3). The difference can serve as a measure of the improvement of our estimate.

### 5.3 An alternative approach: Continuous distributions

A major drawback of the discrete approach is that we need to choose appropriate lengths for the intervals $[n_i, n_{i+1})$. A convenient way to avoid this is to model $N$ by a continuous distribution, say $\mathcal{N}(\mu, \sigma)$. The facts that $\mu$ is large and that many different values $N$ can occur validate this choice. Since negative values of $N$ make no sense, we assume that $\sigma \ll \mu$, so that the probability of such values occurring is negligible. We assume again that the probability of a driver being a user is $\lambda$, in other words, that $U$ is binomially distributed with parameter $\lambda$ and $N$, given $N \in \mathbb{N}$. By the law of large numbers

$$B(N, \lambda) \sim \mathcal{N}(N\lambda, \sqrt{N\lambda(1-\lambda)}).$$

The rest of the analysis follows the previous section, and we shall not give the details.

## 6 The dependence of travel time on road usage

In the previous section, we used an unspecified function $g$ to describe the dependence of the travel time $T$ on the number of users $N$ of a road segment; see (1). In this section we use a simple model of traffic dynamics to arrive at a candidate for $g$.

Many different models of traffic flow can be found in the literature. They may be characterized, crudely, as *microscopic* (individual cars, see references in [Cre79, Smu96]), *mesoscopic* (densities and average speeds over segments a few hundred meters in length, again see [Cre79, Smu96]) and *macroscopic* (on the level of networks, see [RB94]).

We focus on the mesoscopic level. The existing models are too detailed for us, and we make a few extra simplifications.

## 6.1 Traffic model

The model we describe here is a simplified version of the models used in [Cre79, Smu96, vS99a]. The main simplification is that we consider a single road segment, on which we assume that homogeneous conditions prevail.

In section 2 we mentioned the relation between density and equilibrium speed proposed in [Smu96]. We shall assume that traffic flow is always in equilibrium, so that the relation between densities (supplemented by variables describing the road condition) and speed (hence intensity) is deterministic.

Input of the model is a function $A(t)$ describing the influx of cars on the road segment per time unit, in cars per hour. The output is a function $D(t)$ describing the instantaneous density, in cars per kilometre. The density increases due to the influx of cars, and decreases due to the outflux, which is equal to the intensity (in cars/hour). The intensity is a function of the density, namely Smulders' function (see figure 1) multiplied by the density. Denoting the length of the road segment by $L$, this leads to the following model:

$$L\frac{\mathrm{d}D}{\mathrm{d}t} = A(t) - D(t) \cdot S(D(t)) \tag{4}$$

Here $S$ is Smulders' function. In principle, this function depends on various parameters, like road conditions, type of traffic etcetera. For simplicity we shall ignore this and use the following formula:

$$S(D) = \begin{cases} v_{\mathrm{free}}\left(1 - \frac{D}{D_{\mathrm{jam}}}\right) & \text{if } D \leq D_{\mathrm{crit}} \\ v_{\mathrm{free}} D_{\mathrm{crit}}\left(\frac{1}{D} - \frac{1}{D_{\mathrm{jam}}}\right) & \text{if } D > D_{\mathrm{crit}} \end{cases}$$

(See figure 1, and [Smu96, p. 30] for a motivation.) The various parameters are

$$v_{\mathrm{free}} = 110 \text{ km/h}, \quad D_{\mathrm{jam}} = 110 \text{ cars/km}, \quad D_{\mathrm{crit}} = 27 \text{ cars/km}$$

Because we assume that conditions on the road segment are homogeneous, the length of the segment is an important parameter of the model (4). We used the value $L = 30$ km, which led to reasonable results.

## 6.2 Road usage

The influx of cars at a certain instant, per unit of time, is given by $A(t)$. As a model for road usage we take

$$A(t) := N\sqrt{\frac{\alpha}{\pi}}e^{-\alpha t^2}, \tag{5}$$

a bell curve, where $N$ is the total number of cars passing the road, and $\alpha$ is related to the width of the bell curve. Both parameters influence the development of a jam.

## 6.3 Analysis

We are interested in the throughput of the road segment. One statistic related to this is the *average time* it takes to travel through the segment. The time spent is the time of exit minus time of entry; the average time spent on the segment for all cars is therefore

$$\begin{aligned} T &= \frac{\int_{-\infty}^{\infty} t\,(\mathrm{outflux}(t) - \mathrm{influx}(t))\,\mathrm{d}t}{\int_{-\infty}^{\infty} \mathrm{influx}(t)\mathrm{d}t} = \frac{\int_{-\infty}^{\infty} t\,(D(t) \cdot S(D(t)) - A(t))\,\mathrm{d}t}{\int_{-\infty}^{\infty} A(t)\mathrm{d}t} \\ &= \frac{1}{N}\int_{-\infty}^{\infty} -Lt\frac{\mathrm{d}D(t)}{\mathrm{d}t}\mathrm{d}t = \frac{L}{N}\int_{-\infty}^{\infty} D(t)\mathrm{d}t, \end{aligned}$$

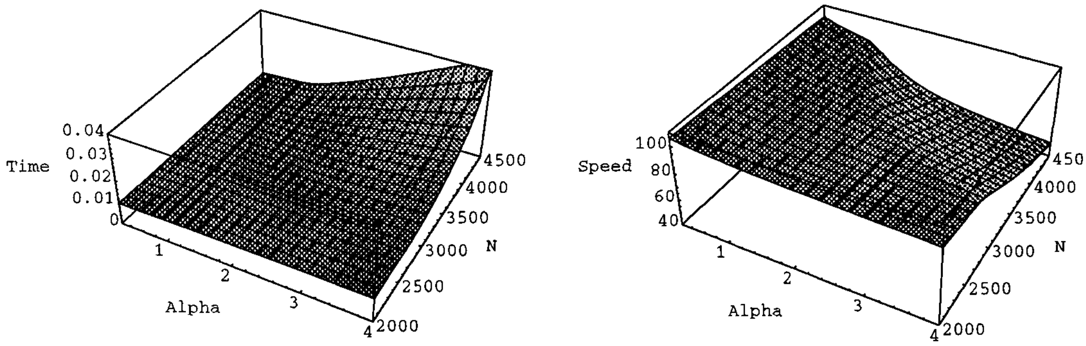

Figure 3: The average travel time, and average speed, on a road described by traffic model (4) under influx (5), for varying parameter values $N$ and $\alpha$.

where we integrated by parts, with vanishing boundary terms. This is a first candidate for the function $g(N)$.

A different but related statistic is the *average speed.* It may be calculated by noting that an intensity of cars $D(t) \cdot S(D(t))$ experience a speed $S(D(t))$; the average speed therefore is

$$V = \frac{1}{N} \int_{-\infty}^{\infty} D(t) \left(S(D(t))\right)^2 \mathrm{d}t$$

For parameter values $N \in [2000, 4500]$ and $\alpha \in [\frac{1}{4}, 4]$, both statistics have been plotted in figure 3. In both plots a 'ridge' can be seen, along which $T$ and $V$ seem to have discontinuous derivatives. The corresponding curve in the $\alpha - N$ plane is related to the onset of jams.

### 6.3.1 Critical curve

We now try to obtain an estimate of this curve of critical parameter values. The pair $(\alpha, N)$ is critical when the density reaches, but not increases beyond, the critical density $D_{\text{crit}}$, at some time $t_{\text{crit}}$. At this moment $\frac{\mathrm{d}D}{\mathrm{d}t} = 0$. Plugging this into (4) we get that $t_{\text{crit}}$ satisfies

$$A(t_{\text{crit}}) = D(t_{\text{crit}}) S(D(t_{\text{crit}})) = D_{\text{crit}} S(D_{\text{crit}}) = I_{\text{crit}}, \tag{6}$$

where the critical intensity $I_{\text{crit}}$ is defined by $I_{\text{crit}} := D_{\text{crit}} S(D_{\text{crit}}) = 27S(27) = 2241$ cars/h. Equation (6) has two solutions, one negative and one positive. Since $D$ lags $A$ and reaches its maximum after $A$ does, only the positive solution is relevant.

The remaining condition is that $D(t_{\text{crit}}) = D_{\text{crit}}$. To solve this equation we need to solve the differential equation (4). To simplify the latter, first note that $0 \leq D \leq D_{\text{crit}}$ globally. In this range $S(D)$ depends linearly on $D$, and varies by approximately 25%. We approximate $S(D)$ by a constant $S_{\text{avg}}$. This constant is chosen somewhere between 110 and 83 km/h, but with a bias towards the lower value since $S(D)$ affects the differential equation more when $D$ is larger. Then (4) becomes linear,

$$L \frac{\mathrm{d}D(t)}{\mathrm{d}t} + S_{\text{avg}} D(t) = A(t),$$

and can be solved by variation of constants, $D(t) = (1/L) \int_{-\infty}^{t} e^{S_{\text{avg}}(u-t)/L} A(u) \mathrm{d}u$. The important parameter here is $S_{\text{avg}}/L$, which in our case is approximately 3. This justifies the use of the following estimate which is more useful for our purpose, and which is valid for large parameter values $S_{\text{avg}}/L$:

$$D(t) = \frac{1}{S_{\text{avg}}} A\left(t - \frac{L}{S_{\text{avg}}}\right) + \frac{L^2}{2S_{\text{avg}}^3} A''\left(t - \frac{5L}{3S_{\text{avg}}}\right) + \cdots \tag{7}$$

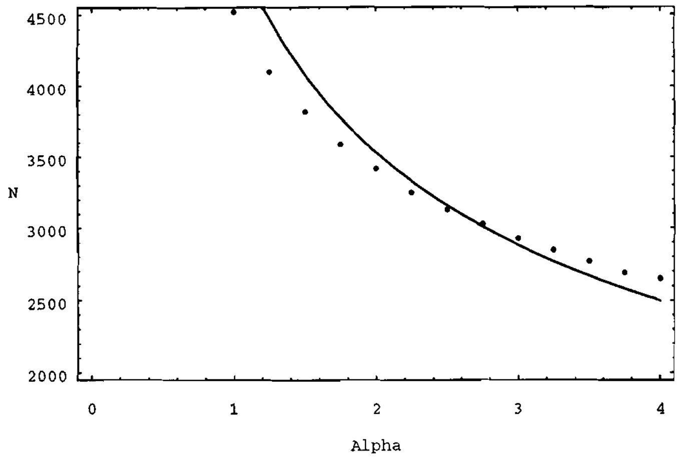

Figure 4: Curve of critical values in the $N - \alpha$ plane: jams start to occur upon crossing this curve from the left. Dots are obtained by numerically solving (4); the curve is the approximation (8),(9).

Using only the first term, we get the following condition for $t_{\text{crit}}$:

$$A\left(t_{\text{crit}} - \frac{L}{S_{\text{avg}}}\right) = S_{\text{avg}} D_{\text{crit}}$$

Expanding to first order, and using (6), we can rewrite this as $A'(t_{\text{crit}}) = (1/L)S_{\text{avg}}(I_{\text{crit}} - D_{\text{crit}}S_{\text{avg}})$. ¿From (5) we find that $A'(t)/A(t) = -2\alpha t$, and using this the condition becomes

$$2\alpha t_{\text{crit}} = \frac{1}{L} S_{\text{avg}} \left( \frac{D_{\text{crit}} S_{\text{avg}}}{I_{\text{crit}}} - 1 \right) \tag{8}$$

Given $\alpha$, we can solve (8) for $t_{\text{crit}}$, and then

$$N_{\text{crit}} = I_{\text{crit}} \sqrt{\frac{\alpha}{\pi}} e^{\alpha t^2_{\text{crit}}} \tag{9}$$

Fitting the resulting curve to the curve obtained by numerical integration, we find good agreement at $S_{\text{avg}} = 95$ km/h. The critical curve is plotted in figure 4.

### 6.3.2 Average time with jams

We now analyze what happens when we cross the critical curve. We assume that the time during which the density exceeds the critical value is small, compared to the total time interval considered. Let $N$ and $\alpha$ be on the critical curve, and let $t_{\text{crit}}$ denote the instant at which critical density is reached, and $I_{\text{crit}} := D_{\text{crit}} S(D_{\text{crit}}) = D(t_{\text{crit}})S(D(t_{\text{crit}}))$ the critical intensity. We increase $A(t)$ by a factor $\epsilon$, that is, we set

$$A(t) := (1+\epsilon)N\sqrt{\frac{\alpha}{\pi}} e^{-\alpha t^2}$$

For small $\epsilon$, the time spent in the jam-regime $D \geq D_{\text{crit}}$ will be small. This justifies approximating $I$ and $A$ by linear models,

$$\begin{aligned} I(D(t)) &= I_{\text{crit}} - \gamma(D(t) - D_{\text{crit}}) \\ A(t) &= (1+\epsilon) I_{\text{crit}} (1 - \beta(t - t_{\text{crit}})), \end{aligned}$$

where we used that $A(t_{\text{crit}}) = I_{\text{crit}}$ when $\epsilon = 0$. Here $\gamma = \frac{\mathrm{d}I}{\mathrm{d}D}$ as $D \searrow D_{\text{crit}}$, and $\beta = -\frac{\mathrm{d}A}{\mathrm{d}t}/I_{\text{crit}}$ at $t = t_{\text{crit}}$. For $\epsilon > 0$ the critical density will be reached for $t_1 < t_{\text{crit}}$. Using $D(t) \approx (D_{\text{crit}}/A(0))A(t - t_{\text{crit}})$ (equation (7)), valid when $D \leq D_{\text{crit}}$, we find the approximation

$$D(t) \approx (1+\epsilon)D_{\text{crit}}\left(1 + \frac{A''(0)}{2A(0)}(t - t_{\text{crit}})^2\right) = (1+\epsilon)D_{\text{crit}}(1 - \alpha(t - t_{\text{crit}})^2)$$

for the case $\epsilon > 0$ and $D \leq D_{\text{crit}}$. Solving $D(t) = D_{\text{crit}}$ we find

$$t_1 = t_{\text{crit}} - \sqrt{\frac{\epsilon}{\alpha(1+\epsilon)}}$$

For convenience we now choose translated variables $t'$ and $D'$, so that $t' = 0$ corresponds to $t_1$, and $D' = 0$ to $D_{\text{crit}}$. Setting up the differential equation for the jammed regime in these variables, we get

$$L\frac{\mathrm{d}D'}{\mathrm{d}t'} = (1+\epsilon)I_{\text{crit}}\left(1 - \beta\left(t' - \sqrt{\frac{\epsilon}{\alpha(1+\epsilon)}}\right)\right) - I_{\text{crit}} + \gamma D' = a - bt' + \gamma D'$$

where $a = I_{\text{crit}}(\epsilon + (1+\epsilon)\beta\sqrt{\epsilon/\alpha(1+\epsilon)})$ and $b = I_{\text{crit}}(1+\epsilon)\beta$. This equation can be solved by variation of constants again, yielding

$$D'(t') = \left(\frac{a}{L}t' - \frac{b}{2L}t'^2\right)e^{t'\gamma/L} \tag{10}$$

The solution is valid for $D' \geq 0$, that is, between $t'_1 = 0$ and

$$t'_2 = 2\sqrt{\frac{\epsilon}{\alpha(1+\epsilon)}} + \frac{2\epsilon}{(1+\epsilon)\beta} \tag{11}$$

We are interested in the value $\int D(t)\mathrm{d}t$. Without jams this would be $N/S_{\text{avg}}$, see (7). With jams the value becomes larger, due to two effects. First of all, from the second term in (11) it is seen that the time interval where $D \geq D_{\text{crit}}$ is longer than it would be without jams. Secondly, the density in this interval is larger than it would be without jams. Integrating (10) over the appropriate interval, and truncating at degree $\epsilon^2$, we the following formula:

$$T = \frac{L}{N}\int_{-\infty}^{\infty} D(t)\mathrm{d}t = \frac{L}{S_{\text{avg}}} + \begin{cases} 0 & \text{if } N < N_{\text{crit}} \\ \dfrac{LD_{\text{crit}}}{N}\dfrac{2\epsilon}{(1+\epsilon)\beta} + \dfrac{16\beta I_{\text{crit}}}{3N}\sqrt{\dfrac{\epsilon^3}{\alpha^3(1+\epsilon)}} & \text{if } N \geq N_{\text{crit}} \end{cases} \tag{12}$$

where $N = (1+\epsilon)N_{\text{crit}}$. It turns out that for reasonable parameter values, the last term, measuring the large-density effect, is the least important one. As a final improvement, we replace the constant $S_{\text{avg}}$ by a number that is 110 for $N = 0$, and linearly decreases to $S_{\text{avg}}$ when $N$ reaches its critical value. The resulting curve, and the numerically obtained statistic $T$, are plotted in figure 5.

We conclude that the expression (12) may serve as a good approximation of $g(N)$ below and around critical densities.

# 7 Conclusions

Ericsson's problem has many different aspects, which makes it impossible to give a precise answer to their question. Instead we have tried to provide an overview of these aspects, which helped us to subsequently formulate and analyze a model problem.

We selected relevant literature and described some of the recent research in the area. Traffic routing problems and traffic density predictions, as well as traffic flow models, have been studied

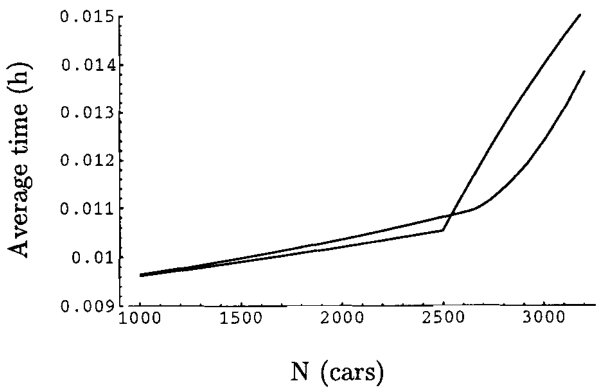

Figure 5: Average time to travel through the segment, for fixed $\alpha = 4$, as a function of $N$. The smooth curve has been determined numerically, the other curve corresponds to (12).

greatly. It turned out that even advance booking had been investigated, an idea which is closely related to Ericsson's ideas. We indicated a few problems that may arise when a CRP is implemented. Our main concerns are, that too few drivers may cooperate for traffic density estimates to improve significantly, that users may give unreliable information or ignore the CRP's advice, and that road conditions (which cannot be predicted well in advance) may influence the traffic densities more than the demand.

To get a sense for the relation between user cooperation and reliability of travel time estimates we modelled a single road segment, under strong assumptions. Here we used the variance of travel time estimates as a measure of reliability. We also analyzed a simplified traffic model to find a relation $T = g(N)$ between road usage $N$ and average travel time $T$. The fraction $\lambda$ of the drivers that use the CRP appears to influence the travel time predictions in two ways. If the number of cars on a road is not near a certain critical number, the variance of the travel time depends more or less linearly on the fraction $\lambda$ (in this case the derivative $g'(N)$ is roughly constant when $N$ changes due to CRP advices). Far from critical situations, the effect of users information will therefore only be noticeable when many drivers become users. If the number of cars on a road is near the critical number, $g'(N)$ changes drastically with small variations of $N$. This may cause a higher order dependence of the variance on $\lambda$, and means that user information becomes more useful. It is however hard to predict in advance whether the situation on a road will be near criticality.

Moreover, a change in $N$, caused by the CRP's advice, may also let $g'(N)$ increase. This can result in a less reliable estimate than the estimate without users information. Partly this is due to our notion of reliability; however, paradoxical situations may occur, related to Braess' observation. It would be interesting to study this, and identify when reduced reliability, or increased travel time, can occur as a result of providing users with better information. Another topic that seems interesting and relevant to study is the dependence of jams on road conditions versus road usage.

# References

[Bra68] D. Braess. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmenforschung*, 12:258–268, 1968.

[Cre79] M. Cremer. *Der Verkehrsfluß auf Schnellstraßen.* Springer Verlag, 1979.

[dR94] E. de Romph. *Dynamic traffic assignment models.* PhD thesis, Technical university Delft, 1994.

[Hof85] Douglas R. Hofstadter. *Metamagical themas.* Basic Books, 1985.

[Kel91] Frank Kelly. Network routing. *Philosophical Transactions of the Royal Society A*, 337:343–367, 1991. Also available at `http://www.statslab.cam.ac.uk/~frank/CP/-CPREAD/nr.html`.

[Mie00] Mark Mieras. Alstublieft reserveren voor een autoritje. Intermediair 3, January 20, 2000.

[Nie00] Niema. `http://www.autoritadvies.nl`, 2000.

[Pay71] H. J. Payne. Models of freeway traffic and control. In *Mathematical models of public systems. Simulation Council Proceedings*, pages 51–61, 1971.

[RB94] B. Ran and D. E. Boyce. *Dynamic Urban Transportation Network Models*, volume 417 of *Lecture Notes in Economics and Mathematical Systems.* Springer-verlag, 1994.

[Sch98] Jan Schreuder. A strategic approach for the ambulance covering of the province of friesland. Memorandum No. 1462, University of Twente, Enschede, 1998.

[Smu96] S. A. Smulders. *Control of freeway traffic flow*, volume 80 of *CWI Tract.* CWI, 1996.

[Tim90] New York Times. What if they closed 42nd street and nobody noticed?, December 1990.

[vS99a] J. H. van Schuppen. Dynamisch routeren. Technical report, CWI, 1999.

[vS99b] J. H. van Schuppen. Implementation of traffic model. Private communication, 1999.

[Wai] Mark Wainwright. A small road network. `http://www.expert.demon.co.uk/mark/-essays/roads.html`.