

MATHEMATICS IN INDUSTRY

scientific proceedings of the 55th

EUROPEAN STUDY GROUP WITH INDUSTRY

Studiegroep Wiskunde met de Industrie 2006

January 30 to February 3, 2006, in Eindhoven

editors: E. R. Fledderus, R. W. van der Hofstad, E. Jochemsz,
J. Molenaar, T. J. J. Mussche, M. A. Peletier, and G. Prokert

December 8, 2006

PREFACE

These are the proceedings of the 55th European Study Group with Industry (Studiegroep Wiskunde met de Industrie), held in Eindhoven from January 30 to February 3, 2006. More than 70 participants attacked six problems, ranging from analysing communication networks to stopping bullets, and from detecting needles to cutting up software.

The proceedings of this week are provided twice, in two different formats. In the current volume the participants provide their own rendering of the week: aimed at a scientific audience, they present the problems, the approach, and the results, in full scientific glory.

In the companion volume, Bennie Mols provides a different view on the week, aimed at a more general audience, and written in Dutch.

Mark Peletier
on behalf of the organization of SWI 2006

The Studiegroep Wiskunde met de Industrie 2006, and these proceedings, were realized with financial support from

the programme Wiskunde Toegepast (NWO/STW)

EURANDOM

Technische Universiteit Eindhoven

Centrum voor Wiskunde en Informatica in Amsterdam

Stichting Industriële en Toegepaste Wiskunde

European Consortium for Mathematics in Industry

CONTENTS

Contents	iii
1 Measure under Pressure	1
1.1 Problem description	1
1.2 The virtual piston model	4
1.3 Simplifications under further assumptions	7
1.4 The Navier-Stokes equations for incompressible fluids	10
1.5 Further side effects	16
1.6 Uncertainty limits	18
1.7 Numerical implementation of Dadson's formula	21
1.8 Recommendations	22
1.9 Bibliography	23
2 Bullet Proof Math	25
2.1 Introduction	25
2.2 Covariate Analysis	27
2.3 General Framework	30
2.4 Generalized Linear Model	31
2.5 Non-Parametric Models	37
2.6 Bootstrap Method	41
2.7 Experimental Set-Up	43
2.8 Conclusions	45
2.9 Bibliography	48
3 Divide and Conquer	51
3.1 Introduction	51
3.2 The general model	52
3.3 A model without capacity restrictions	54
3.4 A model with capacity constraint	56
3.5 A model including waiting times	61
3.6 Pairwise Testing	64
3.7 Conclusions	67
3.8 Bibliography	67

4	Catching gas with droplets	69
4.1	Introduction	69
4.2	Probabilistic approach	72
4.3	Homogenization	76
4.4	Behavior of the penetration time in one dimension	81
4.5	Solution procedure for the stationary problem via conformal mapping	83
4.6	One-dimensional numerical simulation	89
4.7	Two-dimensional numerical simulation	94
4.8	Conclusions	98
4.9	Acknowledgement	100
4.10	Bibliography	100
5	Radioactive Needlework	101
5.1	Introduction	101
5.2	Physical explanations for the artifacts	106
5.3	Medical imaging techniques	109
5.4	Bibliography	114
6	Math Saves the Forest	117
6.1	Introduction	117
6.2	Problem formulation	120
6.3	Probabilistic analysis	122
6.4	Simulations	133
6.5	Conclusions and recommendations	138
6.6	Bibliography	139

MEASURE UNDER PRESSURE

Calibration of pressure measurement

Magdalena Caubergh¹, Jan Draisma², Geert-Jan Franx³,
Geertje Hek⁴, Georg Prokert², Sjoerd Rienstra², Arie Verhoeven²

Abstract

Piston-cylinder assemblies are used to create a calculable pressure in a container, which can then be used for calibration of other instruments. For this purpose one needs to calculate the pressure in the container so accurately that both imperfections in the piston, and the leakage of fluid or gas through the small space between cylinder and piston have to be taken into account. Because of these effects, the piston behaves as if its area was slightly larger than it actually is. This slightly larger area is called the *effective area* of the piston-cylinder assembly, and its computation is the subject of this report.

We derive a formula for this effective area, which under some simplifications leads to the formula used by four European metrological institutes. The formula used by NMI is based on a further simplification. We conclude with some recommendations to NMI concerning which formula to use and how to compute the uncertainty in the results.

KEYWORDS: effective area, piston-cylinder assemblies, pressure balance, thin film approximation.

1.1 Problem description

Six European metrological institutes have compared their respective methods of calculating the effective area of piston-cylinder assemblies, which are used for calibration of pressure measurements [6]. Among them was NMI (Nederlands Meetinstituut = Dutch metrological institute), whose method and results were quite different from those of the other five institutes. NMI asked the study group Mathematics with Industry: first, to explain the differences between the

1: Universiteit Hasselt, 2: Technische Universiteit Eindhoven, 3: Vrije Universiteit Amsterdam, 4: Universiteit van Amsterdam

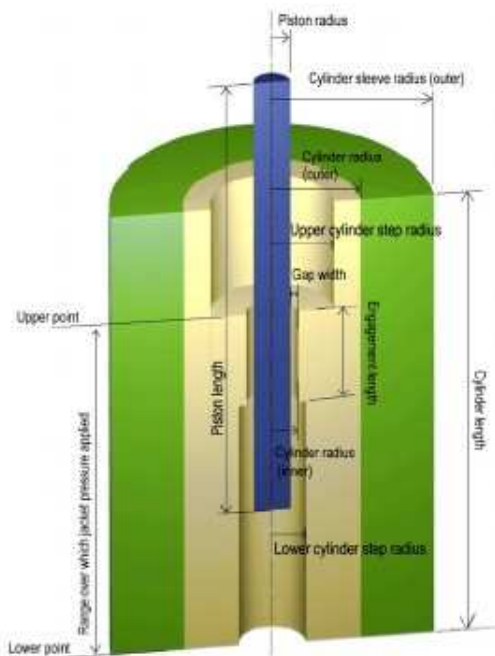


Figure 1.1: Basic geometry of the piston-cylinder [2].

six methods, and second, to recommend a method for computing the effective area. In this note we do this and more: In Section 1.2 we give an introduction to the *virtual piston model* for piston-cylinder assemblies. This model itself is well known and well described in [1], on which most of the remaining sections are based. In Section 1.3 we show how, under certain simplifications, the model yields the various formulas used by the metrological institutes. In Section 1.4 we give a mathematically rigorous treatment of the Navier-Stokes equations for incompressible Newtonian fluids, which also lead to the same formula. Of course, our model itself still depends on certain simplifications, and in Section 1.5 we argue, at least for two of these simplifications, that there is no point in relaxing them, since that would only have higher order effects on the results. In Section 1.6 we comment on the computation of uncertainty limits; in Section 1.7 it is described how the formulas for the effective area should be evaluated in a numerical sound way and finally, in Section 1.8 we present the desired recommendations to NMi.

We start with a simplified description of the piston-cylinder unit used for the pressure measurement. Figure 1.1 shows the basic geometry of this device. It consists of a vessel containing a viscous fluid (air or oil) with a (nearly) cylindrical opening in which a piston can move up and down.

Inside the vessel, the fluid is under pressure $p_1 = p_2 + \Delta p$ where p_2 denotes the ambient pressure outside the device. A pressure measurement is done by the weight of the piston so that an equilibrium is reached between this weight and the forces exerted by the fluid on the piston. The largest part of this force results from the pressure acting from below onto the piston.

Between the piston and the surrounding cylinder, however, there is a narrow interstice in which a small amount of fluid is pressed upward. This leads to a frictional force exerted by the fluid to the flanks of the piston, and this force contributes to counterbalancing the weight of the piston.

The so-called *effective area* A_{eff} of the device is the area which would be needed in an idealized situation to counterbalance the weight W of the piston just from the pure pressure force:

$$A_{\text{eff}} := \frac{W}{\Delta p}.$$

Let l denote the length of the piston. We assume that both the piston P and the surrounding cylinder C are perfectly round, i.e., they are given by

$$\begin{aligned} P &:= \{(x, y, z) \mid z \in (0, L), x^2 + y^2 < r(z)\}, \\ C &:= \{(x, y, z) \mid z \in (0, L), x^2 + y^2 < R(z)\}, \end{aligned}$$

respectively.

Our crucial assumption here is that both R and r have small variations and that their difference $h := R - r$ is small compared to the radii:

$$\varepsilon := \frac{h}{r} \ll 1.$$

In practice, ε is of the order 10^{-4} to 10^{-5} . Hence, in the situation we are interested in, terms which are of order ε^2 (or higher) can safely be neglected.

Note that in Section 1.2 and in [1], a slightly differing approach to the concept of effective area is taken: The concept of a so-called *virtual piston* is introduced, consisting of the actual piston together with an annular column of liquid between the actual piston and the neutral surface between cylinder and piston at which no shear forces act inside the liquid. For this virtual piston, the friction force between the piston and the liquid is an internal force, and there is no need to calculate it explicitly. Now in [1] the effective area is defined as

$$S := \frac{W + w}{\Delta p},$$

where w is the weight of the annular liquid column. In our situation, however, including the gravitation force term in the lubrication equations (see Section 1.4) shows that $w/\Delta p$ is of order ε^2 , therefore no difference to order ε exists between A_{eff} and S . Due to this fact, the different approach taken by SMU in their calculation of the effective area does not lead to essentially different results.

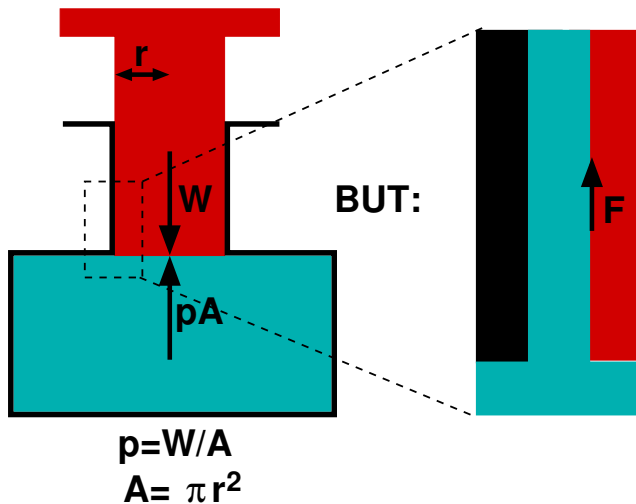


Figure 1.2: A piston-cylinder assembly.

1.2 The virtual piston model

First of all we give a gentle introduction to the *virtual piston model*, using the concept of a *virtual cylinder*. We are given a cylinder and a cylindrical piston of radius r moving in it. The piston has a certain weight W (which includes the so-called applied weights on top of the piston). An ambient buoyancy correction has to be done because of the ambient buoyancy effect on the submerged part of the floating component. This W depends, of course, on the gravity g , but we assume that it can be measured or computed very accurately. In the naive model, depicted on the right in Figure 1.2, the piston and the cylinder are perfect (vertical) cylinders with a perfect fit. In this case, when one knows the area A of the piston, the pressure p can be calculated from the force equilibrium

$$pA = W.$$

Hence it suffices to know, in addition to W , the *nominal area* $A = \pi r^2$ to calculate the pressure p .

However, as suggested on the left in Figure 1.2, there is a small gap between the piston and the cylinder, through which the medium moves upward, exerting an upward frictional force on the piston. Let R be the radius of the cylinder, and set $h := R - r$ and $\epsilon := h/r$. The parameter ϵ will always be assumed small, and in fact our formulas will be exact up to terms of order ϵ^2 . To get rid of this frictional force, one defines the *neutral surface* between the cylinder and piston to be the surface where the velocity of the medium is maximal, and one replaces the piston by the *virtual piston*, which is the actual piston enlarged with the annular column of the medium bounded on the one side by the piston

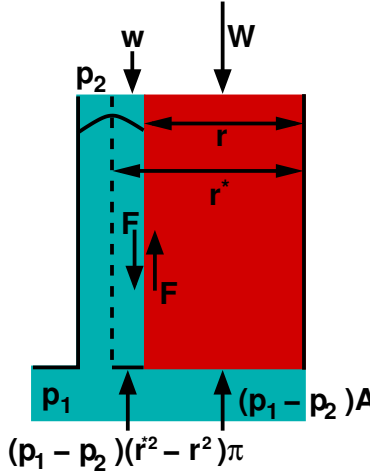


Figure 1.3: The neutral surface and the virtual piston.

and on the other side by the neutral surface—see Figure 1.3. The reason for working with this virtual piston is that no friction is exerted on it anymore: there is no friction among the layers of medium at the neutral surface. Let w be the weight of the annular column of medium between the piston and the neutral surface; again, we assume that w can be measured or calculated very accurately. Now the *effective area* A_{eff} of the piston-cylinder assembly is defined as *the area that would explain why the virtual piston of weight $W + w$ is in equilibrium with the pressure from below*. In a formula, we must have

$$A_{\text{eff}}(p_1 - p_2) = W + w,$$

where p_1 is the pressure below the piston and p_2 is the ambient pressure. Hence, to compute the pressure p_1 it suffices to know W, w, p_2 and A_{eff} .

If the piston and cylinder are still assumed perfect cylinders as in Figure 1.3, then the neutral surface is also a cylinder, whose radius we denote by r^* . It follows from the classical theory of viscous flow between cylindrical surfaces [4] that

$$(r^*)^2 = \frac{R^2 - r^2}{2 \log(\frac{R}{r})}. \quad (1.1)$$

Writing $R = r(1 + \epsilon)$, we get the following expansion for $(r^*)^2$.

$$(r^*)^2 = r^2 \left(1 + \epsilon + \frac{\epsilon^2}{6} + O(\epsilon^3) \right).$$

In fact, r^* is equal to the arithmetic mean $(R + r)/2$ plus terms of order $O(\epsilon^2)$ due to the roundness of cylinder and piston. Other expressions that agree with $(R + r)/2$ up to terms of order ϵ^2 are the geometric mean \sqrt{Rr} or $\sqrt{(R^2 + r^2)/2}$.

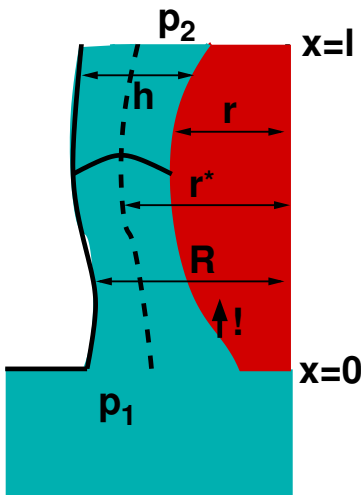


Figure 1.4: A non-perfect assembly.

All these expressions are used in the literature. The next combination of rR and $(R + r)/2$ gives a second order approximation for $\lambda = \frac{1}{6}$.

$$(r^*)^2 = 4\lambda\left(\frac{r+R}{2}\right)^2 + (1 - 4\lambda)rR \doteq r^2(1 + \epsilon + \lambda\epsilon^2). \quad (1.2)$$

In this perfect-cylinder case the formula for A_{eff} is easy:

$$A_{\text{eff}} = \pi(r^*)^2 = \pi((R + r)/2)^2 + O(\epsilon^2).$$

From the six European metrological institutes only NMI uses this formula. However, the piston-cylinder assemblies under consideration are not perfect. We do assume that they have perfect rotational symmetry around a vertical axis (see Model B in Section 1.5 for a discussion of this assumption). Then the piston and the cylinder are described by their radii r and R as a function of the vertical coordinate x ; see Figure 1.4. The neutral surface will also have rotational symmetry, hence be given by its radius r^* as a function of $x \in [0, l]$. Furthermore, the pressure p is a function of x , as well, and so is h . Following [1] we sometimes write r_0, R_0, r_0^*, h_0 for the values of r, R, r^*, h at 0, and p_1, p_2 for $p(0), p(l)$.

Now the virtual piston has weight $W + w$, and this is in equilibrium with the following forces exerted on it:

1. A force equal to $\pi r^*(0)^2 p_1 - \pi r^*(l)^2 p_2$ due to the pressure working on both ends of the virtual piston, and
2. a force equal to $\int_0^l p(\xi) \frac{d\pi(r^*)^2}{dx}(\xi) d\xi$ due to the vertical component of the fluid pressure acting on the inclined flanks of the virtual piston.

Equilibrating these with $W + w$ and partial integration yields

$$\begin{aligned}
 W + w &= \pi r^*(0)^2 p_1 - \pi r^*(l)^2 p_2 + \int_0^l p(\xi) \frac{d\pi(r^*)^2}{dx}(\xi) d\xi \\
 &= \pi r^*(0)^2 p_1 - \pi r^*(l)^2 p_2 + [p(\xi) \pi r^*(\xi)^2]_0^l - \int_0^l \pi r^*(\xi)^2 \frac{dp}{dx}(\xi) d\xi \\
 &= - \int_0^l \pi r^*(\xi)^2 \frac{dp}{dx}(\xi) d\xi
 \end{aligned} \tag{1.3}$$

This formula has a nice intuitive interpretation: the infinitesimal pressure difference $-\frac{dp}{dx}$ at height ξ pushes upward against the circular horizontal cut at height ξ of the virtual piston; and all these forces together are in equilibrium with $W + w$.

Dividing by $p_1 - p_2$, we find that

$$A_{\text{eff}} = -(p_1 - p_2)^{-1} \int_0^l \pi r^*(\xi)^2 \frac{dp}{dx}(\xi) d\xi. \tag{1.4}$$

Now we will often use the geometric mean \sqrt{Rr} as an approximation for r^* . Moreover, we introduce the two new variables $u := r - r_0$ and $U := R - R_0$, which are also assumed to be $O(\epsilon)$. Then $(r^*)^2 = rR + O(\epsilon^2) = (r - u)(R + u) + O(u) + O(\epsilon^2) = r_0(r_0 + h_0 + U + u) + O(u) + O(\epsilon^2)$. Substituting this approximation, we find that the effective area is approximately

$$A_{\text{eff}} \simeq \pi r_0^2 \left\{ 1 + \frac{h_0}{r_0} - \frac{1}{r_0(p_1 - p_2)} \int_0^l (u(\xi) + U(\xi)) \frac{dp}{d\xi} d\xi \right\}. \tag{1.5}$$

Most European metrological institutes use equivalent or simplified versions of this formula. The goal is now, given r and R as functions of x (or rather, lists of their values measured at finitely many levels in $[0, l]$), and assuming a suitable model for the pressure p , to compute the effective area A_{eff} using the formula above.

1.3 Simplifications under further assumptions

Having determined the formula (1.4), it is still not possible to calculate the effective area of the piston: the formula contains the unknown pressure p_1 (which is to be determined!) and, even worse, the derivative $p'(\xi)$ of the pressure in the thin layer between the piston and the cylinder. In this section we show that from formula (1.5) one can, under additional assumptions, derive various other formulas in which all variables are known.

Since the annulus between the cylinder and the piston is very small ($h/r = \epsilon \ll 1$) the fluid motion in this gap is at zeroth order well described by the so-called thin film or lubrication approximation of the Navier-Stokes equation. For the derivation we use (again) the rotationally symmetric nature of the problem

and the fact that the ratio h/r is small. These two features allow us to consider the problem as a 2D one and then apply the rotational symmetry to obtain a full 3D picture. The fluid in a vertical 2D slice has velocity $\mathbf{v} = (v_1, v_2)$, where $v_1(x, y)$ is the velocity component in the vertical x -direction and $v_2(x, y)$ the component in the horizontal y -direction. The equations are then

$$\frac{\partial p}{\partial x} = \mu \frac{\partial^2 v_1}{\partial y^2}, \quad \frac{\partial p}{\partial y} = \mu \frac{\partial^2 v_2}{\partial y^2}, \quad \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} = 0,$$

where μ is the viscosity, that is assumed independent of the pressure p . For a viscous fluid the natural boundary conditions are $v_1 = v_2 = 0$ on the walls, so on $y = 0 + u(x)$ and $y = h + U(x)$. In first approximation this yields the solution $v_2 \equiv 0$, $p = p(x)$, $v_1 = \frac{1}{2\mu} \frac{dp}{dx} (y - u)(y - h - U) \approx \frac{1}{2\mu} \frac{dp}{dx} y(y - h)$ if u and U are much smaller than h .

The fluid velocity flux Q through through a *horizontal* slice of the annulus is the fluid velocity integrated over this area. The rotational symmetry and small fraction h/r yield that this flux is at leading order

$$Q = 2\pi r \int_0^h v_2(y) dy = 2\pi r \frac{1}{2\mu} \frac{dp}{dx} \left[\frac{1}{3} y^3 - \frac{1}{2} h y^2 \right]_{y=0}^h,$$

which yields the formula

$$\frac{Q}{\pi r} = -\frac{1}{6\mu} \frac{dp}{dx} h^3. \quad (1.6)$$

Since the fluid in the annulus is a thin film between two metal side walls, the temperature of the fluid can be assumed constant, so that isothermic laws apply.

Assemblies operating with incompressible fluids

For incompressible fluids, the flux Q is constant. Since r is constant at leading order, this implies that the right-hand side of (1.6) is constant at leading order, so that $\frac{d}{dx} [-\frac{dp}{dx} h^3] = 0$. Integration leads to

$$p(x) = p_1 - (p_1 - p_2) \frac{\int_0^x \frac{1}{h(\xi)^3} d\xi}{\int_0^l \frac{1}{h(\xi)^3} d\xi} \quad (1.7)$$

and

$$\frac{dp}{dx} = -(p_1 - p_2) \frac{\frac{1}{h(x)^3}}{\int_0^l \frac{1}{h(\xi)^3} d\xi}. \quad (1.8)$$

Substitution into (1.4) gives the formula

$$A_{\text{eff}} = \frac{\int_0^l \pi r^*(\xi)^2 \frac{1}{h(\xi)^3} d\xi}{\int_0^l \frac{1}{h(\xi)^3} d\xi}. \quad (1.9)$$

This formula only contains variables that are known by measurements and interpolation between the measured data. It is, under the assumption of pressure-independent viscosity, valid for all values p_1, p_2 . In other words, for incompressible fluids the resulting effective area is pressure-independent, which, of course, is what makes the effective area a useful characteristic of piston-cylinder assemblies! This formula and variations on it are used by IMGC, LNE, PTB, and UME. This seems reasonable for liquid-operated assemblies under not too high pressure, as under low pressure liquids in general behave as incompressible fluids. The formulas below for gas-operated assemblies look similar to (1.9), but are slightly more complicated. In particular, the constant

$$C := \int_0^l \frac{1}{h(\xi)} d\xi \quad (1.10)$$

will appear over and over again, and we will abbreviate it to C .

Gas-operated assemblies

For gas-operated assemblies, and also for liquid-operated assemblies under very high pressure, the assumption of incompressibility is no longer realistic. For such fluids it is no longer the flux Q , but the value $Q\rho$ that is constant, where ρ is the density. From (1.6) we then derive that

$$\frac{Q\rho}{\pi r} = -\frac{\rho}{6\mu} \frac{dp}{dx} h^3$$

is constant at leading order. According to the gas law $pV = mRT$ the quotient p/ρ is constant under isothermic conditions, so that $-\frac{p}{6\mu} \frac{dp}{dx} h^3$ is constant and has zero derivate as well. For pressure-independent viscosity integration now leads to

$$p(x) = \left[p_1^2 - \frac{p_1^2 - p_2^2}{C} \int_0^x \frac{1}{h(\xi)^3} d\xi \right]^{1/2}, \quad (1.11)$$

where C is the constant defined in (1.10); hence

$$\frac{dp}{dx}(x) = -\frac{p_1^2 - p_2^2}{2C} \frac{1}{h(x)^3} \left(p_1^2 - \frac{p_1^2 - p_2^2}{C} \int_0^x \frac{1}{h(\xi)^3} d\xi \right)^{-1/2}. \quad (1.12)$$

This formula can again be substituted in (1.4). The resulting effective area A_{eff} is no longer independent of p_1 and p_2 and can in theory not be determined as long as the pressure p_1 is unknown. However, A_{eff} is in fact just a function of the ratio p_1/p_2 . Under the assumption that $\lim_{x \rightarrow \infty} A_{\text{eff}}(x)$ exists, this means in particular that $\lim_{p_2 \rightarrow 0} A_{\text{eff}}(\frac{p_1}{p_2})$ is independent of the value p_1 : if the assembly is immersed in vacuum, so with $p_2 \rightarrow 0$, the effective area is independent of p_1 .

The institute PTB used the resulting expression for A_{eff} (their formula (3) plugged into their (2)), and then extrapolated for $p_1 - p_2 \rightarrow 0$.

Small applied pressure

The expression for A_{eff} for a compressible fluid has two limits, in which the formula becomes more attractive. If we assume that $p_1 \gg (p_1 - p_2)$, we consider the situation in which the pressure difference is small compared to the pressure p_1 (or p_2). Equivalently, one can consider the limit $p_2 \rightarrow p_1$. If, after substitution of (1.12) into (1.4), this limit is taken, then the result is precisely equation (1.9), the formula that gives the effective area in case of an incompressible fluid.

Thus one can conclude that under small pressure differences *any* fluid, compressible or incompressible, leads to the same effective area. This makes it even more attractive to use this formula and validates the choice of IMGC, LNE, PTB, and UME in a sense.

Large applied pressure

The other limit we take is the limit for large applied pressure, so $p_1 \gg p_2$. Since for compressible fluids A_{eff} is a function of p_1/p_2 , the limit $p_2 \rightarrow 0$ describes this situation. In this limit (1.12) reduces to

$$\frac{dp}{dx}(x) = -\frac{p_1}{2C} \frac{1}{h(x)^3} \left(1 - \frac{\int_0^x \frac{1}{h(\xi)^3} d\xi}{C} \right)^{-1/2},$$

which in turn leads to an effective area

$$A_{\text{eff}} = \int_0^l \frac{\pi r^*(x)^2}{2Ch(x)^3} \left(1 - \frac{\int_0^x \frac{1}{h(\xi)^3} d\xi}{C} \right)^{-1/2} dx. \quad (1.13)$$

In deriving these formulas we implicitly assumed that the linear (isothermic) gas law is still valid for these high pressure conditions. The limit (1.13) thus obtained is useful for gas-operated assemblies with high $p_1 - p_2$. Note that the effective area is (again) independent of the values p_1 and p_2 , but differs from the effective area for the low applied pressure or incompressible case. Note also that it involves computing a double integral, where the bound x of the inner integral is the variable of the outer integral; this makes numerical evaluation of the expression above rather awkward.

1.4 The Navier-Stokes equations for incompressible fluids

After the rather informal approach using the virtual piston model, we will now derive formula (1.9) more rigorously, making precise what simplifications of reality underly the model.

The motion of air or oil between the inner r and outer radius R can be described by the Navier-Stokes equations for incompressible Newtonian fluids, given by [5, 3]

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) = -\nabla p + \mu \nabla^2 \mathbf{v} - \rho g \mathbf{e}_x, \quad \nabla \cdot \mathbf{v} = 0, \quad (1.14)$$

where ρ , \mathbf{v} , p and g denote density, velocity, pressure and the gravitational acceleration. This expression is valid for a uniformly constant viscosity μ . This appears to be a reasonable assumption, as the large heat capacity of the metal cylinder is probably able to absorb any generated heat and to keep the temperature, and thus the viscosity, of the fluid constant.

In view of the geometry of piston and cylinder, we choose cylindrical coordinates (r, ϕ, x) , while v , w , u will denote the r , ϕ , x component of the velocity \mathbf{v} . Note the difference between r and r : the latter is, as always, the radius of the piston as a function of x , while the former is the radial coordinate! The stationary problem becomes in axial, radial and circumferential components

$$\rho \left(v \frac{\partial u}{\partial r} + \frac{w}{r} \frac{\partial u}{\partial \phi} + u \frac{\partial u}{\partial x} \right) = \mu \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 u}{\partial \phi^2} + \frac{\partial^2 u}{\partial x^2} \right) - \frac{\partial p}{\partial x} - \rho g, \quad (1.15a)$$

$$\rho \left(v \frac{\partial v}{\partial r} + \frac{w}{r} \frac{\partial v}{\partial \phi} - \frac{w^2}{r} + u \frac{\partial v}{\partial x} \right) = \mu \left(\frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial}{\partial r} (rv) \right) + \frac{1}{r^2} \frac{\partial^2 v}{\partial \phi^2} + \frac{\partial^2 v}{\partial x^2} - \frac{2}{r^2} \frac{\partial w}{\partial \phi} \right) - \frac{\partial p}{\partial r}, \quad (1.15b)$$

$$\rho \left(v \frac{\partial w}{\partial r} + \frac{w}{r} \frac{\partial w}{\partial \phi} + \frac{vw}{r} + u \frac{\partial w}{\partial x} \right) = \mu \left(\frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial}{\partial r} (rw) \right) + \frac{1}{r^2} \frac{\partial^2 w}{\partial \phi^2} + \frac{\partial^2 w}{\partial x^2} + \frac{2}{r^2} \frac{\partial v}{\partial \phi} \right) - \frac{1}{r} \frac{\partial p}{\partial \phi}, \quad (1.15c)$$

$$\frac{\partial u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (rv) + \frac{1}{r} \frac{\partial w}{\partial \phi} = 0. \quad (1.15d)$$

Both the slowly sinking piston and the rotation can be completely modeled by the boundary conditions! It is convenient to combine p and ρg into the reduced pressure

$$\bar{p} = p + \rho g x. \quad (1.16)$$

When we scale the axial velocity on a typical (as yet unknown) velocity U , the radial velocity on hU/l , the circumferential velocity on the given rotational velocity, say U/δ (where δ is small), radial derivatives on the typical width $h = R - r$, radial distance r and axial derivatives on the slit length l , the circumferential derivatives on a small parameter γ , the (reduced) pressure on $\mu Ul/h^2$, while we call the small parameter $\varepsilon = h/l$ and the Reynolds number in axial direction $Re = \rho U h / \mu$. Notice that $\varepsilon \neq \epsilon = \frac{h}{r}$ but has the same order

of magnitude. Then we get in dimensionless form

$$Re\varepsilon \left(v \frac{\partial u}{\partial r} + \frac{\gamma}{\delta} \frac{w}{r} \frac{\partial u}{\partial \phi} + u \frac{\partial u}{\partial x} \right) = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u}{\partial r} \right) + \gamma^2 \varepsilon^2 \frac{1}{r^2} \frac{\partial^2 u}{\partial \phi^2} + \varepsilon^2 \frac{\partial^2 u}{\partial x^2} - \frac{\partial \bar{p}}{\partial x}, \quad (1.17a)$$

$$Re\varepsilon^2 \left(\varepsilon v \frac{\partial v}{\partial r} + \frac{\varepsilon \gamma}{\delta} \frac{w}{r} \frac{\partial v}{\partial \phi} - \frac{1}{\delta^2} \frac{w^2}{r} + \varepsilon u \frac{\partial v}{\partial x} \right) = \varepsilon^2 \left(\frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial}{\partial r} (rv) \right) + \gamma^2 \varepsilon^2 \frac{1}{r^2} \frac{\partial^2 v}{\partial \phi^2} + \varepsilon^2 \frac{\partial^2 v}{\partial x^2} - \frac{\varepsilon \gamma}{\delta} \frac{2}{r} \frac{\partial w}{\partial \phi} \right) - \frac{\partial \bar{p}}{\partial r}, \quad (1.17b)$$

$$Re\varepsilon \left(v \frac{\partial w}{\partial r} + \frac{\gamma}{\delta} \frac{w}{r} \frac{\partial w}{\partial \phi} + \varepsilon \frac{vw}{r} + u \frac{\partial w}{\partial x} \right) = \frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial}{\partial r} (rw) \right) + \gamma^2 \varepsilon^2 \frac{1}{r^2} \frac{\partial^2 w}{\partial \phi^2} + \varepsilon^2 \frac{\partial^2 w}{\partial x^2} + \gamma \delta \varepsilon^3 \frac{2}{r^2} \frac{\partial v}{\partial \phi} - \frac{\gamma \delta}{r} \frac{\partial \bar{p}}{\partial \phi}, \quad (1.17c)$$

$$\frac{\partial u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (rv) + \frac{\gamma}{\delta} \frac{1}{r} \frac{\partial w}{\partial \phi} = 0. \quad (1.17d)$$

Thus the order of magnitude estimates of the both sides of the equations equal

$$Re\varepsilon, Re\varepsilon\gamma/\delta, Re\varepsilon = 1, \gamma^2\varepsilon^2, \varepsilon^2, 1, \quad (1.18a)$$

$$Re\varepsilon^3, Re\varepsilon^3\gamma/\delta, Re\varepsilon^2/\delta^2, Re\varepsilon^3 = \varepsilon^2, \gamma^2\varepsilon^4, \varepsilon^4, \varepsilon^3\gamma/\delta, 1, \quad (1.18b)$$

$$Re\varepsilon, Re\varepsilon\gamma/\delta, Re\varepsilon^2, Re\varepsilon = 1, \gamma^2\varepsilon^2, \varepsilon^2, \gamma\delta\varepsilon^3, \gamma\delta, \quad (1.18c)$$

$$1, 1, \gamma/\delta = 0. \quad (1.18d)$$

So if ε is small, with $Re \preceq \mathcal{O}(\varepsilon)$, $\gamma^2 \preceq \mathcal{O}(\frac{1}{Re}\varepsilon^4)$, $\delta^2 \succeq \mathcal{O}(Re)$ and $\gamma\delta \preceq \mathcal{O}(\varepsilon^2)$ then all small terms are equal to or smaller than $\mathcal{O}(\varepsilon^2)$, and we are left with

$$\frac{\mu}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u}{\partial r} \right) - \frac{\partial \bar{p}}{\partial x} = 0, \quad (1.19a)$$

$$\frac{\partial \bar{p}}{\partial r} = 0, \quad (1.19b)$$

$$\frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial w}{\partial r} \right) \right) = 0, \quad (1.19c)$$

$$\frac{\partial u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (rv) + \frac{\gamma}{\delta} \frac{1}{r} \frac{\partial w}{\partial \phi} = 0. \quad (1.19d)$$

All this is to be verified a posteriori, because the order of magnitude of U is unknown yet. Equation (1.19a) is the most important equation here, and known as Reynold's lubrication equation. Equation (1.19b) says that the pressure only depends on x . Equation (1.19c) says that circumferential velocity component w is decoupled from the rest of the problem, so it can be ignored as it doesn't contribute to the pressure difference between top and bottom. Equation (1.19d) relates v and w to u , but can also be ignored for the present problem.

The inner cylinder is slowly but steadily moving down by its own weight W , and we assume at time t the bottom to be at height x_p with (constant) velocity u_p , given by

$$x = x_p(t), \quad u_p = \frac{dx_p}{dt}. \quad (1.20)$$

The position of the inner cylinder is conveniently described by

$$r = r(x - x_p). \quad (1.21)$$

The boundary conditions along the cylindrical surfaces $r = r$ and $r = R$, taking into account the same approximation as before by ignoring all $\mathcal{O}(\varepsilon^2)$ -terms, become [7]

$$u = u_p \quad \text{at} \quad r = r(x - x_p), \quad (1.22a)$$

$$u = 0 \quad \text{at} \quad r = R(x). \quad (1.22b)$$

Conservation of mass requires that as much mass is squeezed out of the cavity as corresponds to the incoming volume of the inner cylinder [7]:

$$2\pi \int_r^R u(r, x) r \, dr = -\pi u_p r^2. \quad (1.23)$$

Note that the above expression is the volume flux at height x . This is not the same for every x , because the slit width h may vary with x .

The total force on the inner cylinder [7] is now given by the pressure difference between top and bottom (multiplied by the respective areas) plus the shear and normal stresses of the flow in the slit. Re-expressed in terms of \bar{p} this is given by

$$\begin{aligned} F &= 2\pi \int_{x_p}^{x_p+l} \left[\bar{p} r' + \mu \frac{\partial u}{\partial r} r \right]_{r=r} dx + \pi \rho g \int_{x_p}^{x_p+l} r^2 (x - x_p) dx \\ &\quad + \pi [r^2(0) \bar{p}(x_p) - r^2(l) \bar{p}(x_p + l)] \\ &= \pi \int_{x_p}^{x_p+l} \left[-\frac{d\bar{p}}{dx} r^2 + 2\mu \frac{\partial u}{\partial r} r \right]_{r=r} dx + \pi \rho g \int_0^l r^2(s) ds. \end{aligned} \quad (1.24a)$$

From equations (1.19a, 1.22a, 1.22b) and (1.23) we have

$$2r\mu \frac{\partial u}{\partial r} = r^2 \frac{d\bar{p}}{dx} - \frac{\frac{1}{2} \frac{d\bar{p}}{dx} (R^2 - r^2) + 2\mu u_p}{\log(R/r)} \quad (1.25a)$$

$$\frac{d\bar{p}}{dx} = \frac{4\mu u_p}{(R^2 + r^2) \log(R/r) - (R^2 - r^2)} \quad (1.25b)$$

(Note that velocity u_p is as yet unknown.) This leads to the total force on the inner cylinder to be given by

$$F = -2\pi\mu u_p \int_0^l \frac{R^2 + r^2}{(R^2 + r^2) \log(R/r) - (R^2 - r^2)} ds + \pi \rho g \int_0^l r^2 ds. \quad (1.26)$$

The unknown velocity u_p is obtained from the condition that for a steady situation the force F should be equal to the weight of the cylinder W . So we have

$$u_p = - \frac{W - \pi \rho g \int_0^l r^2 ds}{2\pi \mu \int_0^l \frac{R^2 + r^2}{(R^2 + r^2) \log(R/r) - (R^2 - r^2)} ds}. \quad (1.27)$$

This yields all the information necessary to determine the pressure difference between top and an bottom. If u_p is known it is also possible to estimate the value of U . From (1.23) it follows that $\pi(R^2 - r^2)U \approx \pi|u_p|r^2$, so

$$U = \mathcal{O}\left(\frac{|u_p|}{2\varepsilon}\right).$$

If $|u_p| \preceq \mathcal{O}(\varepsilon^2)$ the previous assumption that $Re \preceq \mathcal{O}(\varepsilon)$ is correct. Because of (1.31a) and (1.31b) we can estimate that

$$u_p \simeq \frac{W}{6\pi\mu l} \varepsilon^3.$$

If we use the following estimates (for air):

$$\begin{aligned} r &= l = 6 \text{ cm}, \\ W &= 5000 \text{ g}, \\ \mu &= 1.78 \cdot 10^{-4} \text{ g/cm s}, \\ \rho &= 1.2 \cdot 10^{-3} \text{ g/cm}^3, \\ \varepsilon &= 5 \cdot 10^{-5} \end{aligned}$$

we obtain

$$\begin{aligned} u_p &= 3 \cdot 10^{-8} \text{ cm/s}, \\ U &= 3 \cdot 10^{-4} \text{ cm/s}, \\ Re &= 6 \cdot 10^{-7}, \end{aligned}$$

so indeed $Re \ll \varepsilon$. The order condition $\gamma^2 \preceq \mathcal{O}(\frac{1}{Re}\varepsilon^4)$ is fulfilled if $\gamma \leq 3 \cdot 10^{-6}$, a very small number. For $\delta \sim 8 \cdot 10^{-4}$ the side-effects can be neglected because then $\delta^2 \succeq \mathcal{O}(Re)$ and $\gamma\delta \preceq \mathcal{O}(\varepsilon^2)$. Because $\delta = U/2\pi r f \approx 7.96 \cdot 10^{-6}/f$, it follows that the rotational frequency $f \sim 10^{-2} \text{ rev/s} = 0.6 \text{ rev/min}$. This result is different from the results of Michels [1], who found much higher critical speeds lying generally within the range 28 to 32 rev/min. This difference could be explained by the fact that we used different parameter values. However, it is also mentioned in [1] that there is evidence that considerably lower speeds are quite practical with well-made piston-cylinder assemblies.

We have

$$\bar{p}(x_p) - \bar{p}(x_p + l) = - \int_{x_p}^{x_p+l} \frac{d\bar{p}}{dx} dx \quad (1.28)$$

Thus we get

$$p(x_p) - p(x_p + l) = \rho gl + \frac{2}{\pi} \left(W - \pi \rho g \int_0^l r^2 ds \right) \frac{\int_0^l \frac{1}{(R^2 + r^2) \log(R/r) - (R^2 - r^2)} ds}{\int_0^l \frac{R^2 + r^2}{(R^2 + r^2) \log(R/r) - (R^2 - r^2)} ds}. \quad (1.29)$$

This is a complete and, within the theory of lubrication flow with slowly varying walls [5] and moderate Reynolds number, exact result. We can make considerable progress, however, by using the fact that the slit is not only slowly varying but also very close to, and very thin compared to, a typical cylinder radius. We choose a fixed radius R_{eff} , which will be chosen in a convenient way and which will correspond to the effective area, and introduce

$$r(s) = R_{\text{eff}} - h_1(s), \quad (1.30a)$$

$$R(s) = R_{\text{eff}} + h_2(s), \quad (1.30b)$$

$$h(s) = h_1(s) + h_2(s), \quad (1.30c)$$

$$R(s) = r(s) + h(s), \quad (1.30d)$$

where h_1 and h_2 are both of the same order of magnitude as h . Then we can approximate for small h

$$(R^2 + r^2) \log(R/r) - (R^2 - r^2) = \frac{2h^3}{3R_{\text{eff}}} + (h_1 - h_2) \frac{h^3}{3R_{\text{eff}}^2} + \mathcal{O}(h^5/R_{\text{eff}}^3), \quad (1.31a)$$

$$R^2 + r^2 = 2R_{\text{eff}}^2 - 2R_{\text{eff}}(h_1 - h_2) + \mathcal{O}(h^2). \quad (1.31b)$$

This yields the rather unwieldy expression

$$p(x_p) - p(x_p + l) \simeq \rho gl + \left(\frac{W}{\pi R_{\text{eff}}^2} - \rho gl + \frac{2\rho g}{R_{\text{eff}}} \int_0^l h_1 ds \right) \frac{\int_0^l \frac{1}{h^3} - \frac{1}{2} \frac{h_1 - h_2}{R_{\text{eff}} h^3} ds}{\int_0^l \frac{1}{h^3} - \frac{3}{2} \frac{h_1 - h_2}{R_{\text{eff}} h^3} ds}. \quad (1.32)$$

A clever choice of R_{eff} , however, is the one which makes

$$\int_0^l \frac{h_1 - h_2}{R_{\text{eff}} h^3} ds = 0. \quad (1.33)$$

This is achieved by

$$R_{\text{eff}} = \frac{\int_0^l \frac{R + r}{h^3} ds}{2 \int_0^l \frac{1}{h^3} ds} \quad (1.34)$$

Notice that R_{eff} can be viewed as the radius of a generalized *neutral surface*. In this case our expression greatly simplifies to

$$p(x_p) - p(x_p + l) \simeq \frac{W}{\pi R_{\text{eff}}^2} + \frac{2\rho g}{R_{\text{eff}}} \int_0^l h_1 ds \quad (1.35)$$

This can be interpreted as the well-known effective area, see [1] and Section 1.1. If we define

$$A_{\text{eff}} = \pi R_{\text{eff}}^2 \quad (1.36)$$

and note that

$$w = 2\pi R_{\text{eff}} \rho g \int_0^l h_1 ds \quad (1.37)$$

is (to the order of approximation) equal to the weight of the cylinder of fluid between R_{eff} and r , then

$$A_{\text{eff}}(p(x_p) - p(x_p + l)) \simeq W + w \quad (1.38)$$

In conclusion: the systematic and most general definition of effective area, for piston-cylinder assemblies operating with incompressible fluids, is given by equations (1.34) with (1.36). Up to order ϵ^2 , this approach leads to the same expression as formula (1.9) in Section 1.3.

1.5 Further side effects

The model (1.4) which has been derived in section 1.2, is based on a lot of assumptions.

- The piston and cylinder are axisymmetric.
- The vertical velocity of the piston is zero in the stationary case.
- The system converges sufficiently fast to the stationary state.
- There is no rotation because the stationary case is stable.
- The piston and cylinder have the same axis.
- The elastic properties of the material of the piston and cylinder are not important.
- The temperature variations because of the friction can be neglected.

In practice these assumptions are not fulfilled, as we now explain. We will shortly describe the physical aspects of the piston-cylinder unit and enumerate the side-effects which are not modelled by Dadson's theory, which is described in Section 1.2.

There is a fluid/gas below the piston and also between the walls of the piston and cylinder. In what follows, we will concentrate on the case of incompressible

fluids. We start from an initial state, for which there is no fluid between the walls. Because of the gravity force the piston will sink rather fast. Fluid will flow between the moving walls, which implies an upward force as reaction on the gravity force. This upward force will grow when the piston sinks until both forces are equal (stationary case). Because this equilibrium should be unstable, the piston is rotating with fixed angular frequency around its fixed axis. Because of the viscosity the cylinder will also rotate. It follows that the walls of the piston and cylinder do not touch each other.

The first side-effect is the fact that the radii of the piston and cylinder depend on z and ϕ . Second, the piston falls with a constant speed in the stationary case. In Dadson's theory it was assumed that this speed is zero but this is not always true. We are interested in the stationary case for which the piston falls with this constant speed and is still rotating. A third side-effect is that the piston is rotating in order to get rid of the instability.

In section 1.4 a general formula for the effective area is given. It is directly derived from the Navier-Stokes equations for incompressible Newtonian fluids. The model includes the fact that the piston is slowly sinking. Furthermore, conditions are given, such that the model can be assumed to be axisymmetric.

In [2, 8] one considers finite element models which also include the elasticity of the material of the piston and the cylinder. If we take care with moving axes we should also consider the dry friction forces if the cylinder and piston touch each other. In [1] it is shown how to deal with the rotation and the moving axes. It has been shown that the resultant of the viscous forces is zero by symmetry.

We will consider the following two extended models.

- A** This model assumes that the piston and cylinder are perfect cylinders around the same axis. There is no rotation. We only consider the effect that the piston slowly sinks in the stationary case.
- B** This model assumes that there is no rotation and it is assumed that the piston does not sink in the stationary case. We only assume that the radius of the piston and cylinder depends on z and ϕ .

Model A: sinking of piston

Consider the piston and cylinder of constant radius r and R . We are interested in the stationary case where the upward wet friction force is equal to the downward gravity force. Note that the force have to be corrected because of the buoyancy force on the piston below the fluid level. For moving axes it has been proved in [1] that the exact value of r^* satisfies

$$(r^*)^2 = r^2 \left(1 + \epsilon + \frac{7}{12} \epsilon^2 + O(\epsilon^3) \right). \quad (1.39)$$

Thus the influence of the sinking piston is $O(\epsilon^2)$.

Model B: variable radius for piston and cylinder

Assume that the radii depend on z and ϕ . Then the neutral surface will also depend on z and ϕ ! It is very hard, to compute $r^*(z, \phi)$ in an analytical way. It is defined as the radius of the virtual cylinder between the piston and cylinder for which the force between adjacent layers of fluid will be zero. This means that there the tangential component of the force is zero! Note that formula (1.4) can be written as

$$A_{\text{eff}}(p_2 - p_1) = \int_0^l \pi(r^*)^2 \frac{dp}{dx} dx.$$

Because now r^* and $\frac{dp}{dx}$ also depend on ϕ we get

$$A_{\text{eff}}(p_2 - p_1) = \int_0^l \int_0^{2\pi} \pi(r^*)^2 \frac{dp}{dx} d\phi dx. \quad (1.40)$$

In [1] it is stated that p satisfies the two-dimensional continuity equation:

$$\frac{\partial}{\partial z} \left\{ h^3 \frac{\rho}{\mu} \frac{\partial p}{\partial z} \right\} + \frac{\partial}{\partial \phi} \left\{ h^3 \frac{\rho}{\mu} \frac{\partial p}{\partial \phi} \right\} = 6 \left\{ U \frac{\partial}{\partial z} (\rho h) + V \frac{\partial}{\partial \phi} (\rho h) \right\}, \quad (1.41)$$

where z and ϕ are the axial and circumferential coordinates and U and V are the relative velocities of the two surfaces in the axial and circumferential directions respectively.

From practice it follows that the non-roundness is of the same order as the measurement errors. This implies that it indeed can be assumed that the piston and cylinder are axisymmetric.

1.6 Uncertainty limits

Standard uncertainty of measurements

In all measurements, we have to deal with measurement errors. These are usually modeled as normally distributed uncertainties $\Delta(x_i)$, that are superimposed to the 'real' values of each measurement x_i . The standard uncertainty of measurement x_i is then defined as the standard deviation of $\Delta(x_i)$, which we denote by $\sigma(x_i)$.

For every measuring instrument, some standard uncertainty of measurement is specified, which can be used to calculate the overall uncertainty of some physical entity A , that is derived from the measurements x_i .

If all measurement errors are independent from each other, we can use the following first order approximation to calculate the overall uncertainty in A :

$$\sigma(A)^2 = \sum_{i=1}^n \left(\sigma(x_i) \frac{\partial A}{\partial x_i} \right)^2$$

The piston measurement uncertainties

The NMI has provided us with piston and cylinder measurement data and their standard uncertainties. The piston diameter was measured at 13 different heights (ξ -coordinates), with a standard uncertainty of 50 nm, which is determined by the standard uncertainty of the measuring equipment. However, the sample standard deviation of these 13 measurements is only 14 nm. Even if there are only small fluctuations in the ‘real’ diameter of the piston, we would expect a sample standard deviation of at least 50 nm. The fact that we find a so much smaller sample standard deviation can not be attributed to chance. No matter what statistical test we apply, we always find p-values smaller than 10^{-10} . The same phenomenon appears in the piston measurements that were performed at all other metrology institutes.

We conclude that the standard uncertainties of both piston and cylinder measurements must have a systematic and a random component. Both components are unknown, but the systematic component is always the same for all measurements, whereas the random components are independent from one another. This kind of situation can occur for instance in mass measurements, where the unknown mass is compared to a standard mass. This standard mass has some unknown deviation from the exact value. All measurements performed with the same standard mass will therefore have a systematic error equal to the deviation of the standard mass.

When confronted with systematic uncertainties, we have to adapt the formula for the first order approximation of the overall uncertainty of some physical entity A :

$$\sigma(A)^2 = \sum_{i=1}^n \left(\sigma(x_i) \frac{\partial A}{\partial x_i} \right)^2 + \left[\sum_{i=1}^n \left(\tilde{\sigma}(x_i) \frac{\partial A}{\partial x_i} \right) \right]^2, \quad (1.42)$$

where $\sigma(x_i)$ is the random (uncorrelated) component and $\tilde{\sigma}(x_i)$ is the systematic component of the uncertainty in the measurement of x_i .

It is also possible to model the uncertainties in a more general way, by introducing certain correlations between every pair of separate measurements. This will however lead to quite complicated mathematical models. Another major drawback of this approach is in the fact that it is very hard (if not impossible) to make good estimates for these correlations.

Numerical calculation of the propagation of measurement errors

In this section we give an example of how the propagation formula (1.42) can be handled in a comprehensive numerical way.

We make two assumptions in the computation of uncertainties. First, we assume to deal with the case of an incompressible fluid. Second, we suppose that the approximation \tilde{S} for the effective area A_{eff} , is obtained by replacing the

integrals by Riemann sums (i.e. the integrand is approximated by a staircase function).

These choices are not very restrictive. Indeed, from the analytic formulas, it is derived in section 1.3 that the formula for incompressible as well as the one for compressible fluids coincide in the limit of small applied pressure (i.e. in case $p_1 - p_2 \ll p_1$, that we are dealing with). Hence the restriction to the case of an incompressible fluid does not imply loss of generality. Concrete, we assume that the effective area is expressed by formula (1.43):

$$A_{\text{eff}} = \pi r_0^2 + \pi r_0 h_0 + \pi r_0 \frac{\int_0^l (u(x) + U(x)) h(x)^{-3} dx}{\int_0^l h(x)^{-3} dx} \quad (1.43)$$

Furthermore, we calculate the uncertainty of measurement when the integrals are approximated by Riemann sums. Although it is a primitive approximation technique, it is the root of most other techniques when approximating integrals. As a consequence, the formula (1.44) derived below, can serve as a first order approximation of the random component of uncertainty of the effective area in general.

The Riemann sum approximation is obtained by dividing the interval $[0, l]$ first in n subintervals of equal length, say $[x_i, x_{i+1}]$, $0 \leq i \leq n-1$ with $x_0 = 0$ and $x_{i+1} - x_i = l/n$. Then, the expression in the right-hand side of (1.43) can be approximated by

$$\begin{aligned} \tilde{S} &= \pi r_0^2 + \pi r_0 h_0 + \pi r_0 \frac{\sum_{i=0}^n (u(x_i) + U(x_i)) h(x_i)^{-3} \frac{l}{n}}{\sum_{i=0}^n h(x_i)^{-3} \frac{l}{n}} \\ &= \pi r_0^2 + \pi r_0 h_0 + \pi r_0 \frac{\sum_{i=0}^n (u_i + U_i) h_i^{-3}}{\sum_{i=0}^n h_i^{-3}} \\ &= \tilde{S}(r_0, R_0, r_1, R_1, \dots, r_n, R_n) \end{aligned}$$

where $u_i = u(x_i)$, $U_i = U(x_i)$, $h_i = h(x_i)$, $\forall 0 \leq i \leq n$; recall also that $h_i = R_i - r_i$, $\forall 0 \leq i \leq n$. Let us denote by $\sigma(y)$ the random component of uncertainty of y and by $\tilde{\sigma}(y)$ the systematic component of uncertainty of y . Then, the random component of uncertainty of the effective area A_{eff} is defined by

$$\begin{aligned} \sigma(A_{\text{eff}})^2 &= \left(\sigma(r) \frac{\partial \tilde{S}}{\partial r_0} \right)^2 + \left(\sigma(R) \frac{\partial \tilde{S}}{\partial R_0} \right)^2 + \sum_{i=0}^n \left(\sigma(r)^2 + \sigma(R)^2 \right) \left(\frac{\partial \tilde{S}}{\partial h_i} \right)^2 \\ &\quad + \left(\tilde{\sigma}(r) \sum_{i=0}^n \frac{\partial \tilde{S}}{\partial r_i} \right)^2 + \left(\tilde{\sigma}(R) \sum_{i=0}^n \frac{\partial \tilde{S}}{\partial R_i} \right)^2. \end{aligned} \quad (1.44)$$

The partial derivatives that are encountered in (1.44) can be computed as follows. Put $\forall 0 \leq i \leq n$

$$C_i = -3\pi r_0 \cdot \frac{\sum_{j=0}^n h_j^{-3} [(u_i + U_i) - (u_j + U_j)]}{h_i^4 \left(\sum_{j=0}^n h_j^{-3} \right)^2}.$$

Then,

$$\frac{\partial \tilde{S}}{\partial r_0} = \pi R_0 + \pi \frac{\sum_{i=0}^n (u_i + U_i) h_i^{-3}}{\sum_{i=0}^n h_i^{-3}} - C_0;$$

$$\frac{\partial \tilde{S}}{\partial R_0} = \pi r_0 + C_0$$

$$\frac{\partial \tilde{S}}{\partial h_0} = \pi r_0 + C_0 \text{ and } \frac{\partial \tilde{S}}{\partial h_i} = C_i, \forall 1 \leq i \leq n;$$

$$\frac{\partial \tilde{S}}{\partial r_i} = -\frac{\partial \tilde{S}}{\partial h_i} \text{ and } \frac{\partial \tilde{S}}{\partial R_i} = \frac{\partial \tilde{S}}{\partial h_i}, \forall 1 \leq i \leq n.$$

If higher order integration methods are used (like Simpson's rule), every term in the Riemann sum will receive its own coefficient and nothing else will change. Therefore, the same approach can still be applied to the calculation of the partial derivatives.

1.7 Numerical implementation of Dadson's formula

For incompressible fluids the formula (1.9) has been shown in section 1.3 to be a proper approximation of the effective area. The following equivalent formulas are used by four institutes [6]. They are all equivalent and can be derived from (1.9).

$$A_{\text{eff}} = \pi r_0^2 \left\{ 1 + \frac{1}{r_0} \frac{\int_0^l \frac{1}{h^2} dx}{\int_0^l \frac{1}{h^3} dx} + \frac{2}{r_0} \frac{\int_0^l \frac{u}{h^3} dx}{\int_0^l \frac{1}{h^3} dx} \right\} \quad (1.45)$$

$$A_{\text{eff}} = \pi r_0^2 \left\{ 1 + \frac{h_0}{r_0} + \frac{1}{r_0} \frac{\int_0^l \frac{u+U}{h^3} dx}{\int_0^l \frac{1}{h^3} dx} \right\} \quad (1.46)$$

$$A_{\text{eff}} = \pi r_0 \left\{ -r_0 + \frac{\int_0^l \frac{r+R}{h^3} dx}{\int_0^l \frac{1}{h^3} dx} \right\}. \quad (1.47)$$

The integrands of the integrals are continuous functions which have to be approximated by use of the measurements. It is possible to create the integrand-functions themselves directly by interpolation or to create the functions $r, R : [0, l] \rightarrow \mathbb{R}^+$ first. Therefore the cylinder and piston radii are measured for $z = z_i$, resulting in the set $\{(R_i, r_i), i = 1, \dots, N\}$. The grid $\{z_i, i = 1, \dots, N\}$ of the length axis of the piston-cylinder unit can be used to control the accuracy of the resulting continuous functions in an adaptive way. If r, R behave very smoothly it is more efficient to use higher order interpolation, while low order interpolation is better for less smooth surfaces. Furthermore linear interpolation could be used in order to conserve the monotonicity.

Each integral can be evaluated with a numerical integration technique, like Newton-Cotes or Gaussian quadrature formulas. The most straightforward numerical integration technique uses the Newton-Cotes formulas (also

called quadrature formulas), which approximate a function tabulated at a sequence of regularly spaced intervals by various degree polynomials. Common Newton-Cotes formulas include the Trapezoidal Rule (Linear), Simpson's Rule (Parabolic) and Simpson's 3/8 Rule (Cubic). If the functions are known analytically instead of being tabulated at equally spaced intervals, the best numerical method of integration is called Gaussian quadrature, which uses non-uniformly spaced grid points. Common Gaussian quadratures include the Gauss-Legendre Formula and the Gauss-Chebyshev Formula. It could be more efficient to use an adaptive grid, which is more dense where $h(x) \approx 0$. Also it can be synchronized with the grid $\{z_i, i = 1, \dots, N\}$ of the measurements in order to minimize the interpolation errors. The Newton-Cotes formulas are less accurate but significantly less complicated to implement.

If we compare the three unscaled formulas we see that no cancellation errors occur. In all cases the denominator can become very small if the clearance h tends to zero. Therefore we have to scale the variables in order to avoid serious trouble because of roundoff errors. Write $h = \epsilon_h \bar{h}$, $u = \epsilon_u \bar{u}$ and $U = \epsilon_U \bar{U}$, where $\bar{h}, \bar{u}, \bar{U}$ are $O(1)$. Then the formulas (1.45), (1.46), (1.47) can be written as

$$A_{\text{eff}} = \pi r_0^2 \left\{ 1 + \frac{\epsilon_h}{r_0} \frac{\int_0^l \frac{1}{\bar{h}^2} dx}{\int_0^l \frac{1}{\bar{h}^3} dx} + \frac{2\epsilon_u}{r_0} \frac{\int_0^l \frac{\bar{u}}{\bar{h}^3} dx}{\int_0^l \frac{1}{\bar{h}^3} dx} \right\}, \quad (1.48)$$

$$A_{\text{eff}} = \pi r_0^2 \left\{ 1 + \epsilon_h \frac{\bar{h}_0}{r_0} + \frac{1}{r_0} \frac{\int_0^l \epsilon_u \frac{\bar{u}}{\bar{h}^3} + \epsilon_U \frac{\bar{U}}{\bar{h}^3} dx}{\int_0^l \frac{1}{\bar{h}^3} dx} \right\}, \quad (1.49)$$

$$A_{\text{eff}} = \pi r_0 \left\{ -r_0 + \frac{\int_0^l \frac{r+R}{\bar{h}^3} dx}{\int_0^l \frac{1}{\bar{h}^3} dx} \right\}. \quad (1.50)$$

Formulas (1.48) and (1.49) have the advantage that the effective area is expressed in the zeroth order term πr_0^2 and two first order corrections. From literature [9] it appears that the piston shape deviations are much smaller than the cylinder shape deviations, which implies that $\epsilon_u \ll \epsilon_U$. Therefore it is recommended to use the scaled formula (1.48).

1.8 Recommendations

In this document we have shown several models for the piston-cylinder unit. First, there is the *virtual piston model* which has a very useful form, which is given in (1.4). The models in [1] and [6] are specific cases of this model. Second, a formula for the effective area has been derived directly from the Navier-Stokes equations for incompressible fluids. Then it is even possible to get exact results. However, we also proved that the formula (1.9) is sufficiently accurate for incompressible fluids, because the error is of order ϵ^2 , where ϵ is a small number.

Clearly one should always make use of the fact that one can *measure* the dimensions of the piston and cylinder with much higher accuracy than can be

achieved in the *production* process. Therefore we recommend NMI to use a more advanced model for the effective area, like the first order approximation (1.9). It is also used by four other European institutes. A sound numerical integration method should be used like described in Section 1.7. Finally we advise to make a distinction between systematic errors and random errors, which makes it possible to get much sharper uncertainty bounds.

1.9 Bibliography

- [1] R.S. Dadson, S.L. Lewis, and G.N. Peggs. *The Pressure Balance: Theory and Practice*. HMSO, London, UK, 1982.
- [2] T.J. Esward, R.C. Preston, and P.N. Gélat. Piston-cylinder pressure balance design with a negligible distortion coefficient. *Meas.Sci.Technol.*, 14:796–806, 2003.
- [3] A.C. Fowler. *Mathematical models in the applied sciences*. Cambridge University Press, 1997.
- [4] H. Lamb. *Hydrodynamics*. Cambridge, University Press, UK, 1932.
- [5] R.M.M. Mattheij, S.W. Rienstra, and J.H.M ten Thije Boonkkamp. *Partial Differential Equations: Modeling, Analysis, Computation*. SIAM, Philadelphia, 2005.
- [6] G. Molinar, M. Bergoglio, W. Sabuga, P. Otal, G. Ayyildiz, J. Verbeek, and P. Farar. Calculation of effective area A_0 for six piston-cylinder assemblies of pressure balances. Results of the EUROMET Project 740. *Metrologia*, 42:197–201, 2005.
- [7] S.W. Rienstra and T.D. Chandra. Analytical approximations to the viscous glass flow problem in the mould-plunger pressing process, including an investigation of boundary conditions. *Journal of Engineering Mathematics*, 39:241–259, 2001.
- [8] N.D. Samaan. *Mathematical modelling of instruments for pressure metrology*. PhD thesis, City University, London, 1990.
- [9] J. Verbeek. NMI-DH350_rad, 2004.

BULLET PROOF MATH

Modeling the proportion of bullets that pass through a vest

Marco Bijvank¹, Maria Caterina Bramati², Leila Mohammadi³,
Fabio Rigat³, Peter van de Ven³, Roel Braekers⁴,
Tetyana Kadankova⁴, Sergei Anisov⁵, Aad Schaap⁶

Abstract

In order to compare different fibers used in bullet proof vests the velocity V_p at which p percent of the bullets pass through the vest is of high importance, in particular when $p = 50\%$. The objective of this research is to find good estimates for V_p . The available data have been analyzed to examine which aspects influence the probability of perforation and have to be taken into consideration to determine V_p . Next, a general framework has been developed in which the notation is introduced. Several approaches are proposed to find good estimates for V_p . All methods are numerically illustrated. We recommend to use smoothing splines. But a logistic model or an isotonic regression approach with linear interpolation performs also well. The paper ends with a new procedure how the data should be gathered to determine V_p .

KEYWORDS: quantile estimation, classification tree, generalized linear models, isotonic regression, smoothing splines, loss function, bootstrap method

2.1 Introduction

Until recently, effective body protection was an uncomfortable compromise between ballistic protection (i.e., bullet proof vests) and restricted freedom of movement. The need for such a compromise was swept away when a new generation of fibers was developed. The modern vests are made using high performance fibers as p-aramids and high density polyethylene. Protective vests are made out of these fibers, using different technologies as multi-layered fabrics or uni-directional laminates.

1: Vrije Universiteit Amsterdam, 2: Université Libre de Bruxelles, 3: EURANDOM, 4: Universiteit Hasselt, 5: Universiteit Utrecht, 6: Teijin Twaron

When a new kind of fiber has been developed a bullet proof vest is made out of it and the quality of the vest is tested. There are two international standards to determine the ballistic performance, the STANAG based standard and the NIJ standard. All commercial vests currently fulfill the high demands required by these international standards.

The STANAG based test is used to determine the bullet speed where fifty percent of the bullets are stopped by the vest. This velocity will be denoted by V_{50} . This is the easiest way to compare the quality of two vests, where a higher V_{50} means a better quality vest. The determination of the V_{50} is rather easy, since the event for a bullet to perforate the vest is equal to non-perforating the vest at this velocity. The result of the STANAG based test is obtained by firing projectiles in a limited speed range. When 3 stops and 3 perforations are registered, the V_{50} is defined as the average of the 6 corresponding speeds.

With the NIJ standard, the highest stop speed is determined under different pre-described conditions. Using a minimum number of shots (6 or 12) within a given speed range, both stop and perforation shots are required. The maximum stop speed is used as the speed where all projectiles are stopped by the vest.

The disadvantages of both methods will be clear; the number of observations is low. This will lead to a result with a limited accuracy. Unfortunately, the error in the estimated values is not part of the methods. Also, the two independent methods are present for obtaining the characteristics of a vest. For practical reasons, it is desired to use one method only. These disadvantages are nowadays widely recognized by fiber producers, vest manufacturers and the end-users.

The objective of this research is to find a robust method for determining V_{50} and a “highest stop speed” in order to perform quality testing or judging further improvements on fibers. The following characteristics for this method must be used:

- multiple shots on one vest and
- one shooting method for obtaining the speed and the 95% confidence interval for a predefined perforation probability p .

The probability of perforation as a function of the projectile speed does not have to be symmetrical. It is however required to use the same function to determine the velocity at which an arbitrary percentage p of the bullets perforates the vest, in particular for p equal to 1%. This velocity is denoted by V_{01} and for a general p by V_p .

Since data obtained by the standard testing methods are not likely to fulfill the requirements for developing a complex method, different sets of ballistic data have been made available. Within each set, a different ballistic vest (fiber and construction) and bullet has been used. Per situation, 7 individual packs (or vests) have been shot 6 times with one speed per pack. The speed range has been selected in such a way that the range from 0% perforations to 100% perforations was fully covered equidistantly. It must be noted that these data

sets have been generated for experimental use only, and that they are not according the current standards.

Kneubuehl [14] proposes a method to estimate V_{50} . First the author groups the data records in intervals based upon the velocity, with an interval length of 5 m/s. For each interval he estimates a perforation probability by dividing the number of perforations by the total number of shots. Next, a cumulative density function of the normal distribution is fitted through the new data points. The result is the perforation probability as a function of the projectile speed. How to fit such a function is explained in more detail in Section 2.4. Based upon the inverse of this function V_{50} is determined.

In toxicology we find studies that are similar to this research. A general introduction can be found in Agresti [1], Agresti [2] and Emmens [8]. In these toxicology studies, the interest lies in determining models to describe the relationship between the probability of reacting to a certain toxic chemical as a function of the given dose of this chemical. More specifically, for different dose levels, the researchers observe whether the dose results in a toxic reaction.

In this paper we will improve the procedure to determine V_p once the data is provided. But we will also design a new test procedure to gather the data that is used to determine V_p . In Section 2.2 we start with an analysis of the available data. A general framework is presented in Section 2.3, in which we also introduce notation and a general set-up to compare different techniques. In Section 2.4 until Section 2.6 we present different techniques to derive a function that maps a velocity on the probability of perforating a particular vest. In Section 2.4 we will use Generalized Linear Models (GLMs), while the techniques discussed in Section 2.5 do not impose a predefined functional form. The last technique to estimate V_p is a bootstrap method. This approach determines V_p based upon a characteristic at this velocity instead of finding the inverse of a function. In Section 2.7 we propose a new procedure to gather the data. In Section 2.8 we compare the different solution techniques and give a conclusion which method we recommend.

2.2 Covariate Analysis

The available data set contains the 42 data records as explained in Section 2.1 for 10 different vest types. Each data record consists of whether the bullet perforated the vest (this is also called the perforation status), the velocity of the bullet which was shot at the vest, the shot number (1 to 6), the vest number (1 to 7) and the vest type. For one particular vest type the data set contains 126 data records and for one only 36 data records.

The objective of the statistical analysis of the data is to provide an overview of the relationships between the perforation probability (also called the response variable) and its four explanatory variables or covariates, i.e. the bullet velocity, the shot number, the vest number and the vest type. The model employed in this analysis explains the observed variability of the data without making any assumption on the physical or chemical mechanisms which might

have played a role in generating the samples. This means that all data records for the different vest types under investigation are used all together in this analysis.

In the analysis the entire sample is modeled according to a classification tree (Breiman *et al.* [4]). This is a semi-parametric statistical model in which the data is partitioned among several subsamples with significantly different perforation probabilities. The data subgroups are defined by a binary tree where the splits are functions of the covariates. For instance, two groups can be obtained by considering the samples with a bullet velocity smaller than 400 m/s and those with a velocity larger than or equal to 400 m/s. Within the latter group, two clusters of data points can be formed by dividing the samples associated to a particular vest type versus those corresponding to all other vest types, and so on. We will refer to the groups of data generated by a given tree structure as its leaves. In this analysis we do not assume any specific distribution on the space of tree structures, whereas within each leaf we model the perforation status as a Bernoulli random variable with a leaf-specific perforation probability.

In order to estimate the tree structure we perform a stochastic search using the probability of the tree given the data as the score function. This is a simulation-based computationally intensive method which evaluates the uncertainty on the specification of the tree structure conditionally on the sample (Chipman *et al.* [5], Chipman *et al.* [6], Denison *et al.* [7], Holmes *et al.* [13]). Given the best tree structure found by the stochastic search, we estimate the leaf-specific perforation probabilities in a Bayesian fashion. In particular, for each leaf we assume a uniform prior perforation probability. By combining this prior with the Bernoulli likelihood we obtain a Beta perforation probability given the samples falling in the leaf. The Beta distribution can be summarized analytically, providing both a point estimate of the leaf-specific perforation probabilities and their confidence intervals.

Figure 2.1 shows the results of the estimation of the tree structure when all data is analyzed. The tree has a total of eight leaves, which cluster the samples as a function of the bullet velocity and shot number. Notice that this tree structure does not depend on the vest type. This surprising result is emphasized in Table 2.1. For each of the four available covariates, the table shows its estimated probability of inclusion in the tree structure. The covariates with the highest predictive power are the bullet velocity and the shot number but the vest type does not appear to discriminate groups of samples with significantly different perforation probabilities.

Finally, the estimated perforation probabilities for the eight leaves of the tree are presented in Table 2.2. The interpretation of a posterior interval is the probability that the parameter lies in the interval equals $(1-\alpha)$ under the chosen model and prior structure and conditionally on one particular data set. It can be noted that the estimated perforation probability appears to be increasing in the bullet velocity. Moreover, at any given bullet speed, the estimated perforation probability of the first shot is lower than of the second shot, which in turn is higher than any of the other shot numbers.

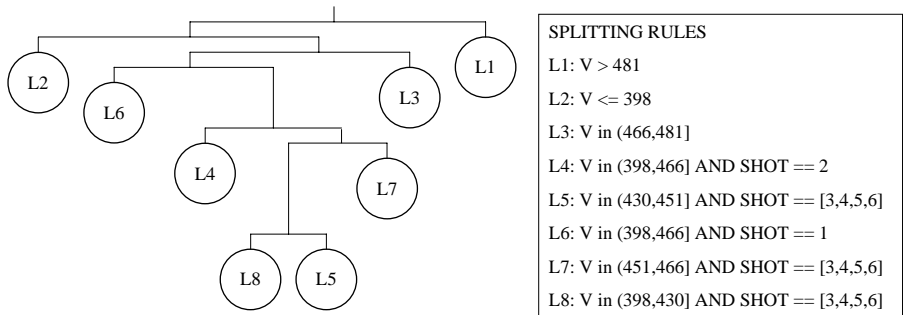


Figure 2.1: The tree structure which fits the data the best.

	covariate			
	velocity	shot number	vest type	vest number
Estimated inclusion probability	1	0.92	0.08	0.02

Table 2.1: The estimated inclusion probabilities for the four covariates to incorporate them into the classification tree.

Leaf number	Estimated perforation prob.	95% Posterior Interval
1	0.95	[0.87; 0.98]
2	0.21	[0.12; 0.35]
3	0.77	[0.65; 0.87]
4	0.60	[0.46; 0.72]
5	0.48	[0.34; 0.63]
6	0.28	[0.17; 0.42]
7	0.25	[0.17; 0.35]
8	0.56	[0.44; 0.69]

Table 2.2: The estimated perforation probabilities including the 95% confidence intervals for the eight leaves of the tree.

Although Figure 2.1 and Table 2.1 make clear that only the velocity and shot number are of relevance to determine the probability of perforation, we consider the velocity and vest type as explanatory covariates in this research. The main reason to take the vest type into consideration is because the performance of vests is required to be compared. The shot number is not used

as covariate, since the shooting pattern should result in independent shots. Apparently this is not the case. Therefore, the shot number is an important parameter which needs to be looked at. But in the remainder of this paper, we will use all data records of one particular vest type to determine V_{50} based upon the speed of the bullet. A motivation not to take the shot number into account is to have a bigger sample set for a fixed combination of covariates. Otherwise there are only 7 data records available per covariate combination.

2.3 General Framework

A procedure has to be developed to determine V_p ; the velocity at which p percent of the bullets perforates the vest. Since we do not take the shot number or the vest number into account (see Section 2.2), the data records of one vest type are presented by (X_i, Y_i) -pairs. The velocity of the i -th shot (expressed in m/s) is denoted by X_i and the event of a perforation by Y_i , where

$$Y_i = \begin{cases} 1, & \text{if shot } i \text{ perforated the vest,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.1)$$

The number of data records is denoted by N , so $i = 1, \dots, N$. Whenever the data set is rearranged into intervals with an interval length of 5 m/s, the (X_i, Y_i) -pairs are transformed into (X'_j, Y'_j) -pairs where $X'_{j+1} = X'_j + 5$. The new response variable Y'_j becomes the average probability of perforating the vest where the velocity of the bullet is in interval j :

$$Y'_j = \left[\sum_{i=1}^N 1_{X_i \in [X'_j - 2.5, X'_j + 2.5)} \right]^{-1} \sum_{i: X_i \in [X'_j - 2.5, X'_j + 2.5)} Y_i,$$

where $1_{\text{condition}}$ is the indicator function:

$$1_{\text{condition}} = \begin{cases} 1, & \text{if } \textit{condition} \text{ is satisfied,} \\ 0, & \text{otherwise.} \end{cases}$$

Figure 2.2 shows the (X_i, Y_i) -pairs as well as the (X'_j, Y'_j) -pairs for the data set where 126 data records are available for a particular vest type.

For every vest type we are interested in finding a function $f(v) : \mathbb{R}_+ \rightarrow [0, 1]$ that maps a velocity v onto the probability of perforating the vest when the bullet has speed v . By taking the inverse $(f^{-1}(p) : [0, 1] \rightarrow \mathbb{R}_+)$ we find V_p . In Section 2.4 and Section 2.5 different approaches are proposed to find an appropriate $f(v)$. A bootstrap method is described in Section 2.6 to determine V_p directly.

In order to compare the different techniques we have to define a measure of fitness that relates the differences between the function $f(v)$ and the data (X_i, Y_i) for $i = 1, \dots, N$. A classical measure of discrepancy is the mean squared error (MSE) as defined in Equation (2.2).

$$MSE = \frac{1}{N} \sum_{i=1}^N [f(x_i) - y_i]^2, \quad (2.2)$$

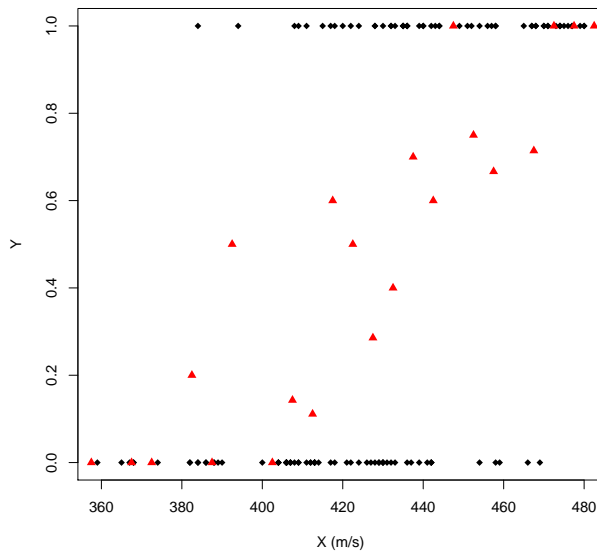


Figure 2.2: The rough data (i.e. the (X_i, Y_i) -pairs) and the rearranged data (i.e. the (X'_j, Y'_j) -pairs) are represented by block dots and red triangles respectively, for one particular vest type.

where (x_i, y_i) are the observed realizations of the stochastic variables X_i and Y_i . Small deviations are not penalized as much as large deviations in this definition for the fitness measure.

2.4 Generalized Linear Model

Generalized linear models (GLMs) are generalizations of the linear model (see McCullagh, *et al.* [15]). In its simplest form, a linear model specifies the linear relationship between a dependent (or response) variable, and a set of predictor variables (or covariates). In this research it is inadequate to describe the observed data (perforation status Y_i) with a linear relationship between the variables (bullet speed X_i). The main reason for this is that the effect of the velocity on the perforation status is not linear in nature.

Link function

In generalized linear models a so-called link function, denoted by g , specifies the connection between the response variable Y_i and the covariate X_i . In this experiment, the response Y_i can take only one of two possible values, denoted

for convenience by 0 and 1 (see also Equation (2.1)). Therefore, we may write

$$P(Y_i = 0) = 1 - \pi_i \quad P(Y_i = 1) = \pi_i \quad (2.3)$$

for the probabilities of non-perforation and perforation respectively. Linear models play an important role in both applied and theoretical work. We suppose therefore that the dependence of Y on X occurs through the linear predictor η_i given by

$$\eta_i = \beta_0 + \beta_1 X_i, \quad i = 1, \dots, N$$

for unknown coefficients β_0 and β_1 . For binary random variables the link function g should map the interval $[0, 1]$ onto the whole real line $(-\infty, \infty)$. So,

$$g(\pi_i) = \eta_i = \beta_0 + \beta_1 X_i, \quad i = 1, \dots, N.$$

A wide choice of link functions is available. Three link functions commonly used in practice are

1. the logit or logistic function

$$g(\pi) = \log(\pi/(1 - \pi)),$$

2. the probit or inverse Normal function

$$g(\pi) = \Phi^{-1}(\pi),$$

3. the complementary log-log function

$$g(\pi) = \log(-\log(1 - \pi)).$$

The first two functions are symmetrical in the sense that

$$g(\pi) = -g(1 - \pi).$$

All three functions are continuous and increasing on $(0, 1)$. This last characteristic is exactly what is required for this research.

To give an example, we look at the logit function

$$\begin{aligned} g(\pi_i) = \log\left(\frac{\pi_i}{1 - \pi_i}\right) &= \beta_0 + \beta_1 x_i \\ \pi_i &= \frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}. \end{aligned}$$

This expression equals $f(v)$ based on Equation (2.3). By inverting this expression we can determine V_p :

$$p = \frac{\exp(\beta_0 + \beta_1 V_p)}{1 + \exp(\beta_0 + \beta_1 V_p)} \Leftrightarrow V_p = \frac{\log\left(\frac{p}{1-p}\right) - \beta_0}{\beta_1}.$$

The same can be done for other link functions. The results are summarized in Table 2.3.

When we combine the probit model with the rearranged data (as described in see Section 2.3) we get the approach proposed by Kneubuehl [14] to determine V_p (see also Section 2.1).

link function	π_i	V_p
logit	$\frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}$	$\frac{\log\left(\frac{p}{1-p}\right) - \beta_0}{\beta_1}$
probit	$\Phi(\beta_0 + \beta_1 x_i)$	$\frac{\Phi^{-1}(p) - \beta_0}{\beta_1}$
complementary log-log	$1 - \exp(-\exp(\beta_0 + \beta_1 x_i))$	$\frac{\log(-\log(1-p)) - \beta_0}{\beta_1}$

Table 2.3: The probability of perforation π_i as a function of the observed velocity x_i for the different link functions. From the inverse of this relationship we get an expression for V_p .

Alternative Predictor

A disadvantage of all three link functions is that the inverses have support on the entire real axis. This means that a velocity of 0 m/s results in a strictly positive probability of perforating the vest. This phenomena is absolutely not true in the experimental setting. A possible solution is to define an alternative predictor η_i as

$$\eta_i = \beta_0 + \beta_1 \log(X_i).$$

For example, the alternative logit function results in

$$\pi = \frac{\exp(\beta_0 + \beta_1 \log(x_i))}{1 + \exp(\beta_0 + \beta_1 \log(x_i))},$$

and

$$V_p = \exp\left(\frac{\log\left(\frac{p}{1-p}\right) - \beta_0}{\beta_1}\right) = \left(\frac{p}{1-p}\right)^{1/\beta_1} \exp(-\beta_0/\beta_1).$$

Having selected a particular model, it is required to estimate the parameters β_0 and β_1 . The parameter estimates are the values that maximize the likelihood function of the observed data. This principle is explained in the next subsection.

Maximum Likelihood

The likelihood of the data is the probability of observing the data for certain parameter values (Ross [18]) and is expressed by Equation (2.4).

$$L(\beta_0, \beta_1; y_1, \dots, y_N) = \prod_{i=1}^N p_{\pi_i}(y_i | \beta_0, \beta_1), \quad (2.4)$$

where $p_{\pi_i}(y_i | \beta_0, \beta_1)$ is the probability of observing y_i when the probability of perforation equals π_i if β_0 and β_1 are the parameter values. From the definition

in Equation (2.3), we should get

$$p_{\pi_i}(y_i|\beta_0, \beta_1) = \begin{cases} \pi_i, & \text{if } y_i = 1, \\ 1 - \pi_i, & \text{if } y_i = 0 \end{cases} \quad (2.5)$$

and, therefore,

$$p_{\pi_i}(y_i|\beta_0, \beta_1) = \pi_i^{y_i} (1 - \pi_i)^{1-y_i}. \quad (2.6)$$

This expression can be substituted into Equation (2.4) to define the likelihood function. A similar expression can be derived when the variable Y'_j is used instead of Y_i .

The objective is to find the values of the two estimators, which maximize the likelihood function. Often it is easier to maximize the log-likelihood function because of its simpler mathematical structure. Therefore, we derive this by taking the natural logarithm of the likelihood function. When we use Equation (2.6) in Equation (2.4) and take the natural logarithm, we get

$$\begin{aligned} l(\beta_0, \beta_1; y_1, \dots, y_N) &= \log L(\beta_0, \beta_1; y_1, \dots, y_N) \\ &= \sum_{i=1}^N \left[y_i \log \left(\frac{\pi_i}{1 - \pi_i} \right) + \log(1 - \pi_i) \right]. \end{aligned}$$

For the logit function, the log-likelihood function equals

$$l(\beta_0, \beta_1; y_1, \dots, y_N) = \sum_{i=1}^N \left[y_i (\beta_0 + \beta_1 x_i) - \log (1 + \exp(\beta_0 + \beta_1 x_i)) \right],$$

which is differentiable in this case. Since the function is concave, the values of β_0 and β_1 that maximize the log-likelihood can be found by solving the first order conditions for the two parameters.

Numerical Results

In Figure 2.3 we show several results of the classical GLMs and alternative GLMs for different link functions. The dots in this figure are the original observed (X_i, Y_i) -pairs (or the rearranged (X'_i, Y'_i) -pairs). The classical GLMs are represented by a solid curve and the alternative GLMs by a dashed curve. We notice that there is not much difference between the two models. However, in the tails of the curves the alternative model always has a lower probability of perforation in the tails of the curves at the same velocity in comparison to the classical models. This is to be expected since the alternative model only allows strictly positive velocities. Therefore, it should have a tighter tail at low velocities and a thicker tail at high velocities.

The maximum likelihood estimators of β_0 and β_1 (e.g., $\hat{\beta}_0$ and $\hat{\beta}_1$ respectively) for each model are presented in Table 2.4, as well as the mean squared error (MSE).

We notice that the value of $\hat{\beta}_1$ is strictly positive in both models. This implies that the curves are strictly increasing. We expected such a result

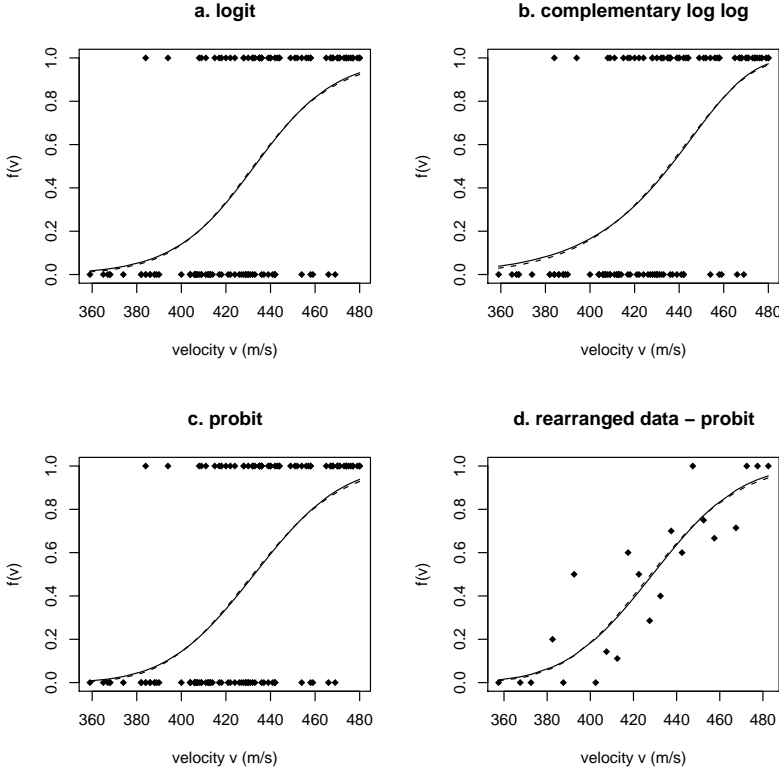


Figure 2.3: The fitted curves for the perforation probability $f(v)$ as a function of velocity v for different link functions. The solid curve is the classical link function and the dashed curve is the alternative link function. The dots are the input data records.

because we know that for an increasing speed of the bullet the probability of perforation will also increase. Based upon these results, we can also conclude that a logit model performs the best (the lowest MSE). However, these are the results for only one vest type. Therefore, we compare all techniques and all data sets (of the different vest types) in Section 2.8.

Based upon the estimated parameter values we determined V_p for the different models with the expressions formulated in Table 2.3. Table 2.5 gives the estimated velocities at which 1% (V_{01}) and 50% (V_{50}) of the bullets perforate the vest. This table also presents a 95% confidence interval for V_p . This interval is generated with the bootstrap method (or resampling): randomly selecting N observations from the data with replacement and obtaining estimates for V_p for the resulting bootstrap sample. We repeated this procedure 1000 times,

model	$\hat{\beta}_0$	$\hat{\beta}_1$	MSE
logit	-23.93	0.0553	0.1650172
alt. logit	-144.2	23.760	0.1650141
probit	-14.09	0.0326	0.1650477
alt. probit	-84.61	13.942	0.1650921
log-log	-16.70	0.0375	0.1659732
alt. log-log	-99.71	16.350	0.1656465
Kneubuehl [14]	-13.52	0.0315	0.1667188

Table 2.4: The likelihood estimators for the different models including the measure of fitness of the model.

calculating estimates for each bootstrap replication. This gives a distribution for the estimate of V_p .

model		estimation	95% Confidence Interval
logit	V_{50}	432.81	[425.04; 441.04]
	V_{01}	349.71	[312.22; 376.21]
alt. logit	V_{50}	432.28	[424.77; 439.77]
	V_{01}	356.27	[327.58; 380.58]
probit	V_{50}	432.74	[424.57; 440.83]
	V_{01}	361.28	[332.69; 384.28]
alt. probit	V_{50}	432.12	[424.15; 440.18]
	V_{01}	365.71	[341.26; 386.59]
c log-log	V_{50}	435.95	[428.00; 443.59]
	V_{01}	322.94	[281.11; 356.47]
alt. c log-log	V_{50}	435.29	[427.42; 442.87]
	V_{01}	335.99	[302.86; 363.45]
Kneubuehl [14]	V_{50}	428.92	[419.90; 437.09]
	V_{01}	355.12	[325.27; 382.83]

Table 2.5: The estimated velocities including their 95% confidence intervals.

For V_{01} we notice that the estimates from the classical models are smaller compared to those from the alternative models. This is not a surprise since we mentioned already that the curves for $f(v)$ show lower values in the tails for the alternative models in comparison to the classical models (see Figure 2.3). Similarly, the confidence intervals in the alternative models are smaller than in the classical models. We would like to mention as well that the confidence interval for V_{01} is wider in comparison to the interval for V_{50} , because there are less data points available around V_{01} .

2.5 Non-Parametric Models

In the previous section we fitted a relationship between the probability of perforation and the velocity by GLMs. However, these techniques place strong assumptions on the shape of this relationship. When we do not want to make such assumptions, we have to fit these curves from the data only. The only restriction we have is that the curve should be monotonic increasing (i.e. non-decreasing). Most of the time, the data does not have this property (see Figure 2.2). Therefore, smoothing has to take place. This can be done in two different ways: either smooth the data first and then find the curve or find a curve on the rough data with the use of smoothing. An example of the first approach is isotonic regression and for the second approach smoothing splines can be used. The third approach we mention in this section is the use of loss functions, which are based upon empirical distributions. All three applications are discussed in this section and we end with numerical results on the three methods.

Smoothing Splines

Splines are piecewise polynomial functions that fit together (Eubank [9]). In particular, for cubic splines, the first and second derivatives are also continuous in every point. Smoothing splines are curves that get reasonably close to the data in a graceful manner such that it gives the appearance of a single curve.

Smoothing splines arise as the solution to the following simple-regression problem: Find the function $\hat{f}(x)$ with two continuous derivatives that minimizes the penalized sum of squares,

$$SS^*(h) = \sum_{i=1}^n [y_i - f(x_i)]^2 + h \int_{x_{\min}}^{x_{\max}} [f''(x)]^2 dx, \quad (2.7)$$

where h is a smoothing parameter (Fox [11]). The first term in Equation (2.7) is the residual sum of squares. The second term is a roughness penalty, which is large when the integrated second derivative of the regression function $f''(x)$ is large. The endpoints of the integral enclose the data. At one extreme, when the smoothing constant is set to $h = 0$ (and if all the x -values are distinct), $\hat{f}(x)$ simply interpolates the data. This function corresponds with the mean squared error, formulated in Equation 2.2. So, small values of h correspond to more emphasis on goodness-of-fit. Conversely, when h is large it places a premium on smoothness. Typically $h \in (0, 1]$. Since we are interested in a monotonically increasing function, we set h to the smallest smoothing parameter such that this restriction is satisfied.

Isotonic Regression

Isotonic regression is a non-parametric method that is used when a dependent response variable is monotonically related to an independent predictor variable

(Barlow *et al.* [3] and Robertson *et al.* [17]). We are indeed looking for an isotonic (i.e., non-decreasing) function where the probability of perforation $f(v)$ depends on the velocity v of the bullet. A commonly used algorithm for computing the isotonic regression is the pair-adjacent violators algorithm (PAVA), which calculates the least squares isotonic regression of the data set (Barlow *et al.* [3] and Robertson *et al.* [17]).

The basic idea of PAVA is the following: sort the (x_i, y_i) -data pairs such that $x_1 \leq x_2 \leq \dots \leq x_N$. If $y_1 \leq y_2 \leq \dots \leq y_N$, then all points are increasing and the algorithm stops. Otherwise, select the first data pair i for which $y_i > y_{i+1}$. In that case replace (x_i, y_i) and (x_{i+1}, y_{i+1}) by their weighted average (x_i^*, y_i^*) , where

$$\begin{aligned} x_i^* &= \frac{w_i x_i + w_{i+1} x_{i+1}}{w_i + w_{i+1}}, \\ y_i^* &= \frac{w_i y_i + w_{i+1} y_{i+1}}{w_i + w_{i+1}}, \\ w_i^* &= w_i + w_{i+1}. \end{aligned}$$

This procedure is repeated until the algorithm terminates. The algorithm starts with weights equal to one ($w_i = 1$ for $i = 1, 2, \dots, N$). The algorithm is applied upon the available data and represented in Figure 2.4.

Now the new data set is such that it is non-decreasing. We can easily find an interpolation scheme to connect the data points and find $f(v)$. We make use of two interpolation schemes in Section 2.5: stepwise interpolation and piecewise linear interpolation.

Loss Function

In this section we describe a method that determines the probability of perforation (or the function $f(v)$) entirely based on empirical distributions. Besides this function $f(v)$, this approach also requires a probability density function of the velocity v , denoted by $g(v)$. Based on the data we can consider the empirical density function of the velocity (denoted by G) and the empirical distribution of $f(v)$ (denoted by F). So,

$$F(x_i) = \begin{cases} 1, & \text{if } y_i = 1, \\ 0, & \text{otherwise.} \end{cases}$$

We want to minimize the result to obtain an estimator for $f^{-1}(p)$. This function is called a loss function (Mohammadi [16]). Select positive α and β and define the loss function as

$$L(a) = \alpha \int_0^a f(v)g(v)dv + \beta \int_a^\infty (1 - f(v))g(v)dv.$$

To minimize L , we take the derivative to a and set it equal to 0:

$$\frac{\partial}{\partial a} L(a) = (\alpha f(a) - \beta(1 - f(a)))g(a) = 0, \quad (2.8)$$

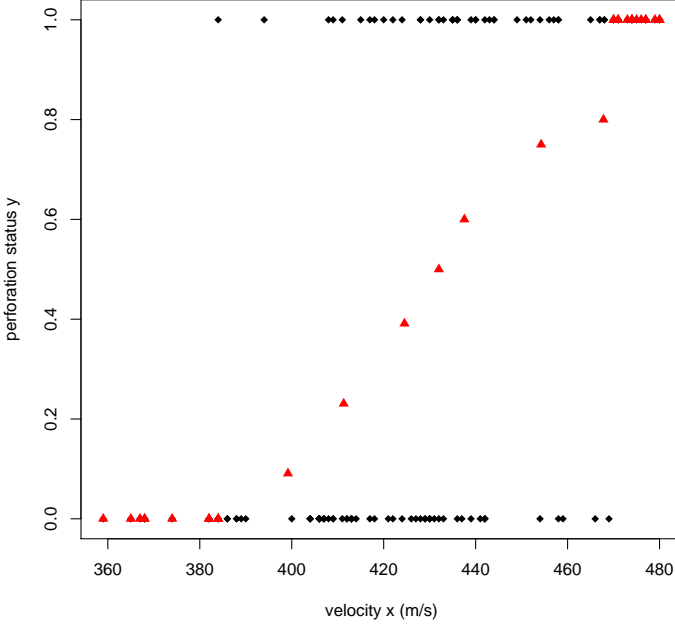


Figure 2.4: We transformed the data from rough data (dots) to monotone increased data (triangle).

Equation (2.8) is solved by a^* with

$$f(a^*) = \frac{\beta}{\alpha + \beta}.$$

Note that

$$\frac{\partial^2 L(a)}{\partial a^2} = (\alpha + \beta)g(a) \frac{\partial}{\partial a} f(a) + (\alpha f(a) - \beta(1 - f(a))) \frac{\partial}{\partial a} g(a),$$

such that

$$\left. \frac{\partial^2 L(a)}{\partial a^2} \right|_{a=a^*} = (\alpha + \beta) \frac{\partial}{\partial a} f(a) \Big|_{a=a^*} g(a^*) \geq 0$$

because $f(v)$ is increasing in v and $\alpha f(a^*) - \beta(1 - f(a^*)) = 0$, using Equation (2.8). It means that a^* is the minimizer of L . We may set $p = \beta/(\beta + \alpha)$. For simplicity, we take $\beta = 1$ and $\alpha = 1/p - 1$. To estimate the inverse of $f(v)$ (i.e., $f^{-1}(p)$), it is now enough to minimize the empirical counterpart of

L , namely

$$\begin{aligned} L(\alpha) &= \left(\frac{1}{p} - 1\right) \int_0^a f(x)g(x)\partial x + \int_a^\infty (1 - f(x))g(x)\partial x \\ &= \left(\frac{1}{p} - 1\right) \sum_{i=1}^n [1_{x_i < a, y_i=1} + 1_{x_i \geq a, y_i=0}]. \end{aligned}$$

Numerical Results

All three non-parametric approaches are implemented and the resulting functions $f(v)$ for each approach are presented in Figure 2.5. The inverses of these functions yield the estimator for V_p . With the use of resampling we constructed a 95% confidence interval (see also Section 2.4). The results are represented in Table 2.6. This table also represents the MSE for each technique.

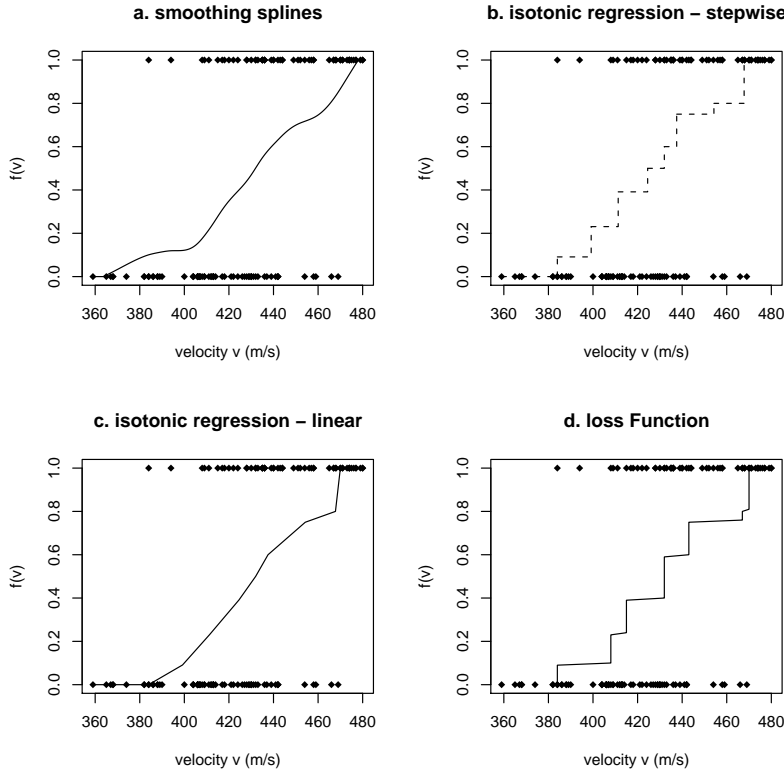


Figure 2.5: The estimates for the perforation probability $f(v)$ as a function of velocity v for different non-parametric approaches. The dots are the input data records.

model		estimation	95% Confidence Interval	MSE
smoothing spline	V_{50}	431.92	[425.02; 440.61]	0.1618964
	V_{01}	365.64	[359.45; 387.52]	
isotonic regression - stepwise	V_{50}	432	[410.82; 440.67]	0.17325
	V_{01}	384	[374; 409]	
isotonic regression - linear	V_{50}	432	[418.27; 442.74]	0.1643158
	V_{01}	385.67	[376.23; 409.12]	
loss function	V_{50}	432	[432; 443]	0.157155
	V_{01}	384	[384; 384]	

Table 2.6: The estimated velocities including their 95% confidence intervals.

2.6 Bootstrap Method

In the previous two sections we were interested in finding a function $f(v)$ in order to determine V_p . In this section we rewrite $f(v)$ as a conditional probability

$$f(v) = P(Y = 1|X = v),$$

the probability of perforation under the condition that the velocity X equals v . Now,

$$P(Y = 1|X = V_p) = p \qquad P(Y = 0|X = V_p) = 1 - p. \quad (2.9)$$

Using Bayes rule, we can rewrite Equation (2.9) as

$$P(Y = 1|X = V_p) = \frac{P(X = V_p|Y = 1)P(Y = 1)}{P(X = V_p)}, \quad (2.10)$$

and

$$P(Y = 0|X = V_p) = \frac{P(X = V_p|Y = 0)P(Y = 0)}{P(X = V_p)}. \quad (2.11)$$

Dividing Equation (2.10) by Equation (2.11) and use Equation (2.9), we get the following expression

$$\frac{p}{1 - p} = \frac{P(X = V_p|Y = 1)P(Y = 1)}{P(X = V_p|Y = 0)(1 - P(Y = 1))}. \quad (2.12)$$

Now we have to compute each of the components of Equation (2.12). Let us first look at $P(Y = 1)$, i.e. the proportion of data records of which the bullet perforates the vest. This can be estimated directly from the data. The two other probabilities can also be derived directly from the data for every observed velocity x_i , $i = 1, \dots, N$, where

1. $P(X = x_i|Y = 1)$ is the relative frequency at which we observe velocity x_i when the vest is perforated, and

2. $P(X = x_i|Y = 0)$ is the relative frequency at which we observe velocity x_i when the vest is not perforated.

Since the property of Equation (2.12) holds for V_p we calculate the ratio

$$\frac{P(X = x_i|Y = 1)P(Y = 1)}{P(X = x_i|Y = 0)(1 - P(Y = 1))},$$

for each observed velocity x_i and the one that is closest to $p/(1 - p)$ is the estimate for V_p .

The main problems with this approach are the few data points in each conditional distribution of the velocity and the fact that we can find different velocities that are closest to the property of V_p . To overcome the first problem we propose to use the bootstrap method to get a distribution for V_p , which allows us to estimate V_p with the average and to construct a 95% confidence interval. The second problem (of multiple velocities satisfying Equation (2.12)) is solved for V_{50} by taking the median and for V_{01} the minimum value of those velocities is selected.

Verification

In order to verify whether the algorithm performs well, we can generate samples from known distributions (like normal or Weibull) that can be used as input. For known distributions, we know what the outcome of the algorithm should be. With the use of a small Monte-Carlo simulation experiment we can test the performance. Table 2.7 shows the deviation of the result from the algorithm with the true outcome for different distributions. The parameters for the distributions are such that the mean and variance are equal to that of the available data set. Based on these results, we can conclude that the algorithm works well for most distributions.

distribution	percentage deviation	
	V_{50}	V_{01}
chi-square	5.35%	6.30%
gamma	5.51%	6.30%
logistic	5.21%	9.17%
log-normal	72.1%	3616.41%
normal	5.34%	6.89%
student	0.33%	1.07%
uniform	5.78%	1.90%

Table 2.7: To verify the bootstrap method, we performed the method with known distributions and therefore the actual outcome is known as well.

Numerical Results

When we apply the proposed procedure to the data set, the resulting estimates for both V_{50} and V_{01} are presented in Table 2.8 including the 95% confidence intervals. Based on these results we conclude that the estimates for V_{50} have a large 95% confidence interval and for V_{01} a rather small interval. This is because the data is collected in a way to determine V_{50} . As a result, not much different velocities are detected satisfying the property as defined in Equation (2.12) for V_{01} .

sample	size	V_{50}		V_{01}	
		estimation	95% Conf. Int.	estimation	95% Conf. Int.
0	126	426.81	[407; 458]	361.90	[359; 368]
1	42	422.03	[412; 429]	398.82	[398; 402]
2	42	458.42	[438; 471]	414.68	[413; 418]
3	42	423.45	[418; 432]	397.94	[397; 404]
4	42	466.59	[448; 483]	434.45	[433; 439]
5	42	459.54	[445; 468]	438.20	[438; 440]
6	42	479.00	[459; 501]	458.60	[458; 460]
7	42	491.87	[471; 499.5]	454.95	[454; 458]
8	42	392.68	[373; 402]	346.68	[346; 351]
9	36	383.88	[361; 406]	350.92	[350; 353]

Table 2.8: Estimates on different samples.

2.7 Experimental Set-Up

The design of the experiment set-up as explained in Section 2.1 is originally developed to determine V_{50} . With the same data statements about V_p for arbitrary p have to be made. Also in other fields where quantile estimation plays an important role (like in toxicology) we see a shift towards generalization. In this section we give some recommendations on the design of future experiments.

The median ($p = 50\%$) is the most commonly used measure of characteristic of the response curve. In some situations this estimation is of intrinsic interest, but more often it is because this quantile is the easiest to estimate (Wu [20]). Recently, several designs have been proposed for estimating quantiles where $10\% \leq p \leq 90\%$ (Wu [20], Stylianou and Flournoy [19]). The designs that are typically suggested are so-called adaptive or sequential designs where the velocity for a run is based on the response (perforation or no perforation) in the previous run(s). Except in the extreme tails of the quantile response function, the optimal design for estimating a particular quantile is a one-point design at the (unknown) target quantile (Ford *et al.* [10]). Hence, a good adaptive strategy should result in taking relatively much observations around the velocity V_p of interest. An adaptive strategy, that has been shown to work

fine for values of p between 10% and 50% is discussed in the next section. We end this section with addressing the problems that arise when this probability of interest is small.

Adaptive Design

Stylianou and Flournoy [19] proposed an adaptive design called the up-and-down Biased Coin Design (BCD). Such a design is such that you tend to be where V_p is. The speed of a random walk, and the mean drift of the random walk, is equal to 0 at V_p , and otherwise the drift is towards V_p . Giovagnoli and Pintacuda [12] showed that the BCD is optimal within a large class of generalized up-and-down biased coin designs in the sense that the distribution of the velocities considered in the experiment is most peaked around V_p .

The BCD procedure is as follows. Before the experiment start, a collection of velocities of interest $\Omega = \{v_1 < v_2 < \dots < v_K\}$ is set. The target velocity V_p should be in the range of Ω . In the first experiment a bullet is shot at velocity $v \in \Omega$. The velocity v may be fixed (e.g. the velocity that is thought to be closest to the target value V_p) or random. If the bullet perforated the vest, the next velocity to shoot with is one slower from Ω . However if the bullet did not perforate the vest, the procedure randomizes: Since we only consider cases where $p \leq 50\%$, the velocity becomes higher according to Ω with probability $p/(1-p)$ and with probability $(1-2p)/(1-p)$ the same velocity is used in the next shot. Appropriate adjustments need to be made at the lowest and highest velocities in Ω .

Small perforation probabilities

Not much is known about the design of experiments when the percentage p is smaller than 10%. A major problem is that the response is binary, which means that the amount of information that we gather each run is very small. Most of the bullets fired at velocities around V_p will be stopped for small values of p . However, some perforations for velocities around V_p are needed in order to estimate the probability of perforation at these velocities and eventually to help locating velocity V_p . Let us denote N_p^r as the number of shots fired at the vest with velocity V_p until the r -th perforation occurs. Under the assumption that bullets are fired independently, this random variable has a negative binomial distribution with parameters p and r . The probability distribution function is given by Equation (2.13).

$$P(N_p^r = n) = \binom{n-1}{r-1} p^r (1-p)^{n-r}, \quad n \geq r. \quad (2.13)$$

The expectation and the variance of N_p^r equals

$$E[N_p^r] = \frac{r}{p}, \quad Var[N_p^r] = \frac{r(1-p)}{p}.$$

Table 2.9 gives some details of the distribution of N_p^r for some small perforation probabilities p and different values of r . When the number of experiments is limited to 150 (or even 500) it will be difficult to locate the velocities V_p for small values of p using only the binary response. Using the depth of the perforation (e.g. the number of perforated layers) as a response variable may be a better way to gather more information from a single shot and to reduce the total number of shots required to determine V_p for small values of p .

p	r	$E[N_p^r]$	$\sqrt{\text{Var}(N_p^r)}$	95% Confidence Interval
5.0%	1	20	4.36	[1; 59]
	3	60	7.55	[3; 124]
1.0%	1	100	9.95	[1; 299]
	3	300	17.23	[3; 628]
0.1%	1	1000	31.61	[1; 2995]
	3	3000	54.75	[3; 6294]

Table 2.9: Some statistics of the negative-binomial distribution of N_p^r for different values of perforation probabilities p and number of perforations r .

2.8 Conclusions

In this paper we investigated the factors that influence the probability that a bullet perforates a bullet proof vest. Section 2.2 made clear that the velocity of the bullet and how many times a vest is shot are most important. However, independence between the different shots has to be assumed in order to satisfy the constraint to use all the shots fired at a vest. We recommend to look into this phenomenon and investigate the shooting pattern that is used. The data analysis also showed that the vest type is of less importance. This makes sense, since it must be noted that all different vests used in this study are constructed for a comparable level of protection. This could well cause the observed absence of influence of the vest.

Comparing Techniques to Estimate V_p

In the remainder of the paper we investigated the relationship between the velocity v and the probability of perforation for every vest type. In particular we have developed several procedures to determine the velocity at which p percent of the bullets go through the vest (denoted by V_p). In the different methods a function $f(v)$ is established which describes this relationship.

When we would like to compare the different approaches that are proposed in this paper, we use the mean squared error as measure of fitness (see Section 2.3). This measure can be computed for all data sets corresponding with

different vest types. In total there are ten data sets. Table 2.10 shows the performance of the model proposed by Kneubuehl [14] (see Section 2.1).

sample	size	Kneubuehl
0	126	0.166719
1	42	0.127233
2	42	0.162312
3	42	0.068530
4	42	0.145247
5	42	0.131702
6	42	0.120194
7	42	0.113723
8	42	0.121351
9	36	0.163257

Table 2.10: The MSE of the model proposed by Kneubuehl [14] for the different vest types (or data sets).

The same can be done for the parametric approaches (GLMs) and the non-parametric approaches, presented in Table 2.11 and Table 2.12 respectively. In order to retrieve one number for the performance of a method, we looked at the deviation of each MSE with the lowest MSE of each sample and averaged this over all samples. The results are shown in Table 2.13.

sample	logit	alt. logit	probit	alt. probit	c log-log	alt. c log-log
0	0.165017	0.165014	0.165048	0.165092	0.165973	0.165647
1	0.126806	0.127032	0.127494	0.127774	0.125250	0.125344
2	0.161179	0.161406	0.161132	0.161374	0.159999	0.160042
3	0.068192	0.068333	0.069782	0.069937	0.068547	0.068493
4	0.141198	0.140954	0.141223	0.140913	0.144903	0.144382
5	0.131761	0.131928	0.132066	0.132292	0.131089	0.131081
6	0.121305	0.121472	0.120707	0.120875	0.119120	0.119242
7	0.111732	0.110691	0.114492	0.113428	0.123853	0.122826
8	0.117431	0.117986	0.118233	0.118817	0.112520	0.112991
9	0.155910	0.153498	0.164483	0.160903	0.179272	0.175300

Table 2.11: The MSE of the classical and alternative GLMs for the different vest types (or data sets).

Based on these results we see the smoothing spline technique to have the lowest average percentage deviation from the lowest MSE. Smoothing splines tend to perform better around the data points. Therefore, other techniques have to be considered as well. Loss functions seem to work well, but the confidence intervals are not convincing. The logistics model (logit model) and

sample	smoothing spline	isotonic regr. (stepwise)	isotonic regr. (linear)	loss function
0	0.161896	0.173250	0.164316	0.157155
1	0.120854	0.154894	0.136643	0.127269
2	0.151425	0.212950	0.160480	0.167148
3	0.069995	0.099286	0.070823	0.064706
4	0.128712	0.137205	0.153609	0.155883
5	0.123416	0.190476	0.137557	0.148669
6	0.117319	0.160788	0.118012	0.135566
7	0.089776	0.090624	0.097533	0.090461
8	0.111329	0.125800	0.104811	0.111609
9	0.145758	0.160601	0.161300	0.140037

Table 2.12: The mean squared error as deviation measure from the real data for the different non-parametric approaches.

technique	average deviation (%)
Kneubuehl	12.78%
logit	8.94%
alt. logit	8.77%
probit	10.22%
alt. probit	9.97%
comp. log-log	11.45%
alt. comp. log-log	40.77%
smoothing spline	2.15%
isotonic regression (stepwise)	26.61%
isotonic regression (linear)	8.83%
loss function	8.01%

Table 2.13: The average deviation as percentage of the lowest MSE for each vest type

the isotonic regression approach with linear interpolation perform also well. Especially when the confidence interval is of interest, we recommend the later two techniques.

The final technique we developed is a bootstrap method in which a particular characteristic at V_p is determined based upon conditional probabilities. A disadvantage of this procedure is that it will only work nicely for particular values of p ($p = 1\%$ and $p = 50\%$ work fine). This procedure will probably give the same results for $p = 1\%$ and $p = 10\%$, since there is not much data available in the region of these particular V_p values. This is not likely to happen in reality.

Experimental Design

The experimental data sets provided for this study are not optimal (equidistant in speed). First we recommend to use different data records in order to determine V_p for different values of p . The data records to determine V_p should concentrate on the influence of the velocity on the perforation probability around V_p . More specifically, we propose a Biased Coin Design, that has been proven to work well in practice for values of p between 10% and 50%. If, however, V_{01} is required to be estimated, this design does not produce a good data set since only perforations of the vest or no perforations are monitored. Other information, like the number of perforated layers, could improve the results. Otherwise, the number of experiments to perform becomes more than a thousand.

2.9 Bibliography

- [1] A. Agresti. *An introduction to categorical data analysis*. Wiley, New York, 1996.
- [2] A. Agresti. *Categorical Data Analysis*. Wiley, New York, 2nd edition edition, 2002.
- [3] R.E. Barlow, D.J. Bartholomew, J.M. Bremner, and H.D. Brunk. *Statistical inference under order restrictions: the theory and application of isotonic regression*. Wiley, London, 1972.
- [4] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Chapman and Hall, New York, 1984.
- [5] H.A. Chipman, E.I. George, and R.E. McCulloch. Bayesian cart model search. *Journal of the American Statistical Association*, 93(443):935–947, September 1998.
- [6] H.A. Chipman, E.I. George, and R.E. McCulloch. Bayesian treed generalized linear models. *Bayesian Statistics*, 7, 2003.
- [7] D.G.T. Denison, B.K. Mallick, and A.F.M. Smith. A bayesian cart algorithm. *Biometrika*, 85, 1998.
- [8] C.W. Emmens. The dose/response relation for certain principles of the pituitary gland and of the serum and urine of pregnancy. *Journal of Endocrinology*, 2:194–225, 1940.
- [9] R.L. Eubank. *Spline smoothing and nonparametric regression*. Dekker, New York, 1988.
- [10] I. Ford, B. Torsney, and C.F.J. Wu. The use of a canonical form in the construction of locally optimal designs for nonlinear problems. *Journal of the Royal Statistical Society, Ser. B*, 54:569–583, 1992.

- [11] J. Fox. *Non-parametric simple regression: smoothing scatterplots*. Sage, Thousand Oaks, 2000.
- [12] A. Giovagnoli and N. Pintacuda. Properties of frequency distributions induced by general ‘up-and-down’ methods for estimating quantiles. *Journal of Statistical Planning Inference*, 74(1):51–63, October 1998.
- [13] C.C. Holmes, D.G.T. Denison, and B.K. Mallick. Bayesian partitioning for classification and regression. *Technical Report, Imperial College London*, 1999.
- [14] B.P. Kneubuehl. Ballistischer schuts. Technical report, 2004.
- [15] P. McCullagh and J.A. Nelder. *Generalized Linear Models*. Chapman & Hall, London, 2nd edition, 1989.
- [16] L. Mohammadi. *Estimation of thresholds in classification*. PhD thesis, University of Leiden, 2004.
- [17] T. Robertson, F.T. Wright, and R.L. Dykstra. *Order restricted statistical inference*. John Wiley & Sons, New York, 1988.
- [18] S.M. Ross. *Introduction to Probability Models*. Academic Press, San Diego, 8th edition, 2003.
- [19] M. Stylianou and N. Flournoy. Dose finding using the biased coin up-and-down design and isotonic regression. *Biometrics*, 58(1):171–177, 2002.
- [20] C.F.J. Wu. Efficient sequential designs with binary data. *Journal of American Statistical Association*, 80:974–984, 1985.

DIVIDE AND CONQUER

Optimizing the release policy of software versions

Philippe Cara¹, Isaac Corro Ramos^{2,3}, Tammo Jan Dijkema⁴,
Lusine Hakobyan³, Markus Heydenreich^{2,3}, Ellen Jochemsz³,
Tim Mussche³, Mihaela Popoviciu-Draisma⁵, Jacques Resing³

Abstract

In this paper we try to find the optimal number of partitions to be made in a piece of software. A model is made for the time-to-market, with respect to which this number is optimized. Refinements are made in this model, taking into account capacity constraints and waiting times. Also, a suggestion is made to use pairwise testing.

KEYWORDS: software partitioning, release policy, time-to-market, testing strategy

3.1 Introduction

ASML, located in Veldhoven, is one of the world's largest producers of lithography systems. Its customers are chip manufacturers, including large companies such as Intel. The chip market is a market with very specific demands. In these times of rapid technological development, it is extremely important to be fast in following new developments on the market. Being the first to offer some feature gives ASML a large advantage over its competitors.

The problem that we are presented with comes from the software department of ASML. They want to keep the software of all machines up to date (the software is such that all machines run on the same software). Currently, ASML issues about 3 new releases of the entire software per year, each with a time-to-market (TTM) of 9 months. Here, time-to-market is defined as the time that elapses between the decision of making a new software release and issuing the tested software to the market. Such a monolithical release includes

1: Vrije Universiteit Brussel, 2: EURANDOM, 3: Technische Universiteit Eindhoven, 4: Universiteit Utrecht, 5: Universität Basel

both bug fixes and new features (Some bug fixes are also delivered in the form of patches, which can be applied cheaper. However, we will not consider this form of updating in this paper.) Installing a new release is very costly for the machine owners. Bringing a machine down could easily cost thousands of dollars per hour. Therefore, some customers choose not to install a new release if there is no urgent reason for it. ASML still supports all older releases.

This form of updating is undesirable for some clients. Suppose a client wishes one new feature. It requests the feature to ASML, which will start implementing it. The update will only be possible in the next release of the entire software, about 9 months away. Also, when the new software is issued to the customer, it comes with all kinds of other features—and possibly bugs.

An alternative is to split the software into a number of pieces, which we will call modules (we were asked to assume this is possible, see e.g. [2]). If a new feature is limited to one module, clients wishing this feature can immediately install the new module once it is released. Other clients can wait longer and install multiple modules at once at a convenient time.

A disadvantage for the software department is that they have to test the new module in a number of environments. Some customers will have the latest version of all other modules installed, but others may still have an old version of another module. Simply requiring all customers to have the latest version of everything is not an option here. What we will require is that all customers have some recent version of all modules (where ‘recent’ will mean something like ‘at most one year old’).

It is easy to see that this approach will lead to an increased testing effort; in principle the number of tests will grow exponentially with the number of modules. However, some customers are happier, because the new feature will be available to them earlier.

The question ASML asks is:

What is the optimal number of modules to split the software into, such that the time-to-market is minimal, while the amount of work remains below some upper bound?

In this paper, we focus on various aspects of this problem. In Section 2, our general model is defined. Next we look at some computational results in this model in the case without capacity constraint in Section 3. In Section 4 we extend our model to take into account a certain capacity of the company that cannot be exceeded. In Section 5 we consider a model including waiting times. Finally we look at some ways of pairwise testing to reduce the cost of testing a new module against older versions of other modules in Section 6. We give some concluding remarks in Section 7.

3.2 The general model

The decision to update a software module will be made by the management based on customer requests. Possibly, by the time of the decision, all development resources are already in use and the development of the new release is

delayed. However, except for Section 3.5, we will assume that the waiting time is 0. This seems reasonable since when a decision to update a software module is made, the current workload of the development resources can be taken into account.

Our aim is to derive a model for the time-to-market when splitting the monolithic software into k pieces. First we want to introduce and discuss the model parameters on a general level, and later make assumptions about these parameters and understand how they influence the outcome.

The proportional size c_j of a module. We want to analyse how the mean time-to-market of the software modules behaves, if we split up the software into k modules. The proportional size of module $j \in \{1, \dots, k\}$ is denoted by c_j , where $\sum_{j=1}^k c_j = 1$. We leave open the meaning of size, one could take, e.g., the number of functionalities.

The development time d_j of a module. We denote the development time of the monolith by D . After splitting the monolith into k modules, the development time of a module is modelled proportional to the size of the module. So we have

$$d_j = Dc_j.$$

The testing time t_j of a module. We denote by T the testing time of the monolith. A new version of a module j will need two kinds of tests prior to the release. The first one will check all the new features of the module in combination with the latest versions of the other $k - 1$ modules. The duration of this test is proportional to the size of module j . A second test will verify whether the new version of module j is compatible with all supported versions of the other modules, except the configuration tested previously. The duration of a such test is denoted by A . Thus

$$t_j = Tc_j + A \left(\prod_{i \neq j} l_i - 1 \right),$$

where l_i is the number of supported versions of block i .

For the time-to-market TTM_j of a module j we thus have

$$\text{TTM}_j = Dc_j + Tc_j + A \left(\prod_{i \neq j} l_i - 1 \right).$$

The number of supported versions l_j of a module. During the compatibility test, ASML tests whether the new release of a software module is compatible with the last l_j versions of the other modules. Hereby l_j is chosen such that all software issued in the last year is supported. Denoting the number of module releases per year by r and writing f_j for the probability that a randomly chosen update request concerns the j -th module, we obtain

$$l_j = \text{Max}\{1, f_j r\}.$$

We imposed the restriction $l_j \geq 1$, because we want to support at least the newest configuration for every software module, even if it is not updated every year.

Thus we finally obtain

$$\mathbb{E}(\text{TTM}(k)) = \sum_{j=1}^k f_j \left(Dc_j + Tc_j + A \left(\prod_{i \neq j} \text{Max}\{1, f_i r\} - 1 \right) \right) \quad (3.1)$$

for the expected time-to-market with k modules.

3.3 A model without capacity restrictions

Choosing model parameters

ASML plans to split the monolithic software into modules of about the same size. Thus, $c_j = 1/k$ for all modules j . The development time of a module j will then be $d_j = D/k$, and the testing time of a new version of a module, in which all the new features of the module are checked, will be T/k . Thus (3.1) simplifies to

$$\mathbb{E}(\text{TTM}(k)) = \frac{D}{k} + \frac{T}{k} + A \sum_{j=1}^k f_j \left(\prod_{i \neq j} \text{Max}\{1, f_i r\} - 1 \right). \quad (3.2)$$

Based on the experience from the monolithic approach, we further assume the following:

- The development time of the monolith is $D = 180$ days.
- The testing time of the monolith is $T = 70$ days.
- A compatibility test needs $A = 2$ days.

The mean time-to-market for different values of k depends on the number of updated modules per year r and on the proportion f_j of update requests that goes to module j . Note that both r and f_j depend on k . Depending on the choice of these functions, the mean time-to-market may change significantly. We will provide calculations for specific choices of these parameters. However, these assumptions need to be checked carefully when validating the model.

The number $r = r(k)$ of module releases per year. Here we write $r(k)$ instead of r to emphasize the k -dependence. Currently the company releases each year about 3 new versions of the monolith. As a first guess, the linear function

$$r(k) = 3k \quad (3.3)$$

seems a good candidate. However, it is rather unclear whether a new version of the monolith would give new features to each of its k submodules. In particular,

due to the extra work needed for the compatibility tests, we expect $r(k)$ to behave sublinear. Nevertheless, (3.3) provides a useful upper bound.

From discussions with ASML representatives we understood that $r(k) = 3k$ might be realistic though. To understand the impact of this choice, we contrast (3.3) with the concave function

$$r(k) = 3k^\beta, \quad (3.4)$$

where $0 < \beta < 1$.

The proportion f_j of update requests that go to module j . The request probabilities for different modules of the monolith are unknown, though it is expected to be rather uneven distributed. That makes it hard to find a pertinent probability distribution for our model. We assume here that the update requests for the k modules are distributed according to the Zipf's law, i.e., a module j will be requested by the customers with the probability f_j , where

$$f_j = \frac{j^{-\alpha}}{\sum_{i=1}^k i^{-\alpha}}, \quad (3.5)$$

in which we take $\alpha = 0.7$. Zipf's law is observed in many applications, e.g. access of web pages or keyword usage in a search engine. For more information on modeling internet traffic using Zipf's law, including technical aspects, we refer to Cunha et al. [7]. The value of $\alpha = 0.7$ as a model for web requests has been suggested by Breslau et al. [4]. Interestingly, Zipf's law was originally used as a model in philology [13].

Computational results

Case $r = 3k$. Using equation (3.2) together with (3.3) and (3.5), we obtain for $r = 3k$ and $\alpha = 0.7$ that $\mathbb{E}(\text{TTM}(k))$ achieves its minimal value for $k = 3$:

$$\mathbb{E}(\text{TTM}(3)) = 96.7.$$

If we split the monolith into 3 modules, the mean time-to-market for the release of one module would be 96.7 days.

Here we have that

- the most popular module would have 4.3 new versions per year;
- the second one would have 2.6 new versions per year;
- the last one would have 2 versions per year.

Comparing $96.7 \times 3 = 290.1$ with $D + T = 250$, we see that the splitting requests a supplementary volume of work equivalent to approximately 40 days. These are the compatibility tests. Depending on which module is updated, there will be necessary maximum 11, respectively minimum 5 compatibility tests. To overcome the problem, the testing resources could be extended, or a model with capacity restrictions as in Section 3.4 could be considered.

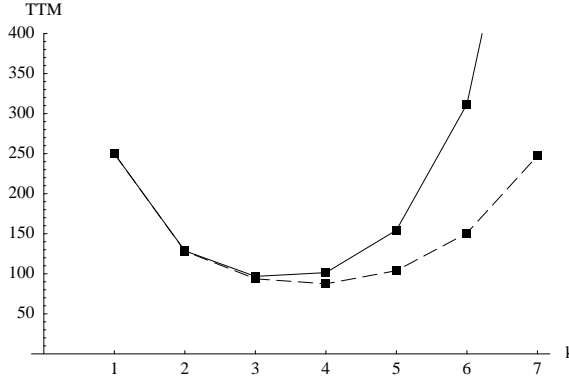


Figure 3.1: The mean time-to-market $\mathbb{E}(\text{TTM}(k))$ in days for $\alpha = 0.7$, and $\beta = 1$ (solid) or $\beta = 0.9$ (dashed).

Case $r = 3k^{0.9}$. In the case $\beta = 0.9$, the minimal time-to-market is achieved for $k = 4$ and

$$\mathbb{E}(\text{TTM}(4)) = 87.5.$$

The mean time-to-market for a module would be 87.5 days. Now

- the most popular module would have 4.2 new versions/year;
- the second one would have 2.6 new versions/year;
- the last two would have 2, respectively 1.6 versions/year.

The supplementary volume of work would be equivalent to about 12 days, but the number of cross tests increases considerably: depending on which module is updated, there are maximum 23, respectively minimum 11, cross tests necessary.

3.4 A model with capacity constraint

Theory

In the last section, we have not yet taken into account the capacity constraint of ASML. Let us assume that the total effort of developing and testing the different modules must stay within the current capacity of the company. The current capacity can be taken as 3 times the total time needed for one new release of the monolith (since at the moment, all ASML's machines and people are working on 3 releases of the full software program per year). This means that the available capacity is

$$\text{cap} = 3(D + T).$$

As before, we assume that we know f_j , the proportion of feature requests for module j , and r , the total number of feature requests per year. So to make every customer happy, we should have a new release each time there is a request. This would be $f_j r$ per year for module j . However, to stay within our capacity the management should decide to put an upper limit M on the number of releases for one module. This means that, in case there are many requests for a certain module, we won't release new modules at each request but rather have M releases of that module per year. So we put

$$\begin{aligned} r_j &:= \# \text{ releases of module } j \text{ per year} = \text{Min}\{f_j r, M\}, \\ \tilde{f}_j &:= \text{proportion of releases of module } j = \frac{r_j}{\sum_{i=1}^k r_i}, \\ M &:= \text{maximal number of releases of each module per year.} \end{aligned}$$

We can now divide our modules into three groups, namely very popular modules, medium popular modules and least popular modules (with respect to feature requests). If we order the modules according to r_j (from high number of releases to low number of releases), we get

Module:	1	...	m	$m+1$...	n	$n+1$...	k
$f_j r \in$	$(3, \infty)$...	$(3, \infty)$	$(1, 3]$...	$(1, 3]$	$(0, 1]$...	$(0, 1]$
$r_j =$	M	...	M	$f_j r$...	$f_j r$	$f_j r$...	$f_j r$
$l_j =$	r_j	...	r_j	r_j	...	r_j	1	...	1

The idea behind this is to release as many versions of the less and medium popular modules as are requested (which per module is at most the current 3 releases per year), and spend the time that is left on releasing M versions of each popular module per year. One can immediately see that this can only give an advantage to the current approach when $M > 3$ is within reach.

So let us compute for a given k , the maximal M to satisfy the capacity constraint. The total time needed to develop and test a new release of module j is given by

$$\text{TTM}_j = \frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right),$$

where A is the (small) testing time needed to test the new version of module j against the previous versions of all modules. Since the total time spent on the r_j releases of module j per year is $r_j \text{TTM}_j$, the total time that we spend on all releases of all modules per year is

$$\sum_{j=1}^k r_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right) \right).$$

Notice that M appears in the product $\prod_{i \neq j} l_i$. Hence, for a given k , we can find the maximal value of M such that

$$\sum_{j=1}^k r_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right) \right) < 3(D + T).$$

For this optimal value of M , we can determine the mean time-to-market for a module

$$\mathbb{E}(\text{TTM}(k)) = \sum_{j=1}^k \tilde{f}_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right) \right),$$

and compare these for the different values of k to see which choice for k is best.

Examples

In practice, the outcome of the analysis will, of course, depend on the constants D , T , and A , for which we will for now substitute $D = 180$, $T = 70$ and $A = 2$. But most importantly, it will depend on the ‘popularity rate’ f_j of the modules, which ASML should evaluate thoroughly before making a decision. The two examples that we consider in this section are

- The f_j are distributed according to Zipf’s law, and r (the total number of requests when splitting the program into k parts), is considered to be $r = 3k$:

$$f_j r = \frac{j^{-\alpha}}{\sum_{i=1}^k i^{-\alpha}} \cdot 3k, \text{ where } \alpha = 0.7 \text{ (see Section 3.3).}$$

- The f_j are distributed according to a ‘toy example’, that originates from the fact that for $k = 5$, ASML can give a guess for a suitable approximation of $f_j r$:

$$f_1 r = 12, f_2 r = 12, f_3 r = 1, f_4 r = \frac{1}{2}, f_5 r = \frac{1}{3}.$$

In the first example, we have

k	$f_1 r$	$f_2 r$	$f_3 r$	$f_4 r$	$f_5 r$	$f_6 r$	$f_7 r$	$f_8 r$	$f_9 r$	$f_{10} r$
1	3	—	—	—	—	—	—	—	—	—
2	3.71	2.29	—	—	—	—	—	—	—	—
3	4.33	2.66	2.01	—	—	—	—	—	—	—
4	4.88	3.01	2.26	1.85	—	—	—	—	—	—
5	5.39	3.32	2.50	2.04	1.75	—	—	—	—	—
6	5.87	3.61	2.72	2.22	1.90	1.67	—	—	—	—
7	6.32	3.89	2.93	2.29	2.05	1.80	1.62	—	—	—
8	6.75	4.15	3.13	2.56	2.19	1.93	1.73	1.57	—	—
9	7.16	4.41	3.32	2.71	2.32	2.04	1.83	1.67	1.54	—
10	7.55	4.65	3.50	2.86	2.45	2.16	1.93	1.76	1.62	1.51

As explained, we will divide the modules for a given k into three groups, the very/medium/least popular modules. The first group, namely the one for which $f_i r > 3$, consists of typically one, two or three modules. For these most popular modules, we will fix the number of releases per year at M , so the expected number of releases of the modules will be

k	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}
1	M	—	—	—	—	—	—	—	—	—
2	M	2.29	—	—	—	—	—	—	—	—
3	M	2.66	2.01	—	—	—	—	—	—	—
4	M	M	2.26	1.85	—	—	—	—	—	—
5	M	M	2.50	2.04	1.75	—	—	—	—	—
6	M	M	2.72	2.22	1.90	1.67	—	—	—	—
7	M	M	2.93	2.29	2.05	1.80	1.62	—	—	—
8	M	M	M	2.56	2.19	1.93	1.73	1.57	—	—
9	M	M	M	2.71	2.32	2.04	1.83	1.67	1.54	—
10	M	M	M	2.86	2.45	2.16	1.93	1.76	1.62	1.51

Assuming $l_j = \text{Max}\{1, r_j\}$ we can determine the maximal M for each k to satisfy

$$\sum_{j=1}^k r_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right) \right) < 3(D + T).$$

We obtain the following maximal values of M :

k	2	3	4	5	6	7	8	9	10
M	3.55	3.26	2.46	1.77	1.17	0.76	0.93	0.71	0.53

One can see that only $k = 2$ or $k = 3$ might be an improvement on the current $k = 1$. Suppose we split in two modules, and we schedule 3.50 releases of the most popular module per year, and 2.29 releases of the least popular module.

Then the mean time-to-market is

$$\begin{aligned} \mathbb{E}(\text{TTM}(2)) &= \sum_{j=1}^k \tilde{f}_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} l_i - 1 \right) \right) \\ &= \sum_{j=1}^2 \frac{r_j}{\sum_{i=1}^k r_i} \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} \text{Max}\{1, r_i\} - 1 \right) \right) \\ &= \frac{3.50}{5.79} \cdot \left(\frac{70}{2} + \frac{180}{2} + 2 \cdot (2.29 - 1) \right) \\ &\quad + \frac{2.29}{5.79} \cdot \left(\frac{70}{2} + \frac{180}{2} + 2 \cdot (3.55 - 1) \right) \approx 129 \text{ days.} \end{aligned}$$

So, while staying within the current capacity, it is possible to split into two modules such that the time that elapses after the management has asked for an update of a module is about 4 months on average.

If we split in three modules, we find

$$\begin{aligned}
 \mathbb{E}(\text{TTM}(3)) &= \sum_{j=1}^3 \frac{r_j}{\sum_{i=1}^k r_i} \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} \text{Max}\{1, r_i\} - 1 \right) \right) \\
 &= \frac{3.26}{7.93} \cdot \left(\frac{70}{3} + \frac{180}{3} + 2 \cdot (2.01 \cdot 2.66 - 1) \right) \\
 &\quad + \frac{2.66}{7.93} \cdot \left(\frac{70}{3} + \frac{180}{3} + 2 \cdot (2.01 \cdot 3.26 - 1) \right) \\
 &\quad + \frac{2.01}{7.93} \cdot \left(\frac{70}{3} + \frac{180}{3} + 2 \cdot (2.66 \cdot 3.26 - 1) \right) \approx 95 \text{ days.}
 \end{aligned}$$

We conclude that for a popularity rate that is Zipf-distributed, $k = 3$ is optimal, as in the previous chapter where the capacity constraint was not taken into account.

Now let us take a look at the second example, the so-called 'toy example'. Recall that this is the case where the popularity rate f_j of the modules is not distributed according to Zipf's law, but that we have

$$f_1 r = 12, f_2 r = 12, f_3 r = 1, f_4 r = \frac{1}{2}, f_5 r = \frac{1}{3}.$$

In other words, we assume that ASML can create 5 modules of approximately the same size, such that one has to be changed once every three years, one has to be changed once every two years, one has to be changed once a year, and the two most popular modules need changes every month.

Now that we have a good approximation of the number of requests in the case that ASML splits the monolith into five parts, can we use that to conclude something in comparison with other values of k ? It seems logical to assume that for $k = 2$, it is possible to create one module that has to be changed once a year, and one module that still has to be changed 12 times a year (because to make two modules out of the five proposed by ASML, we would take the first two together with a part of the third so there will be requests to change this big module once every year). For $k = 3$, $k = 4$, or $k = 6$, it is harder to say something reasonable, because we cannot guess how the number of changes per year for each module would be distributed. So let us compare $k = 1$, $k = 2$, and $k = 5$ for this 'toy example':

k	r_1	r_2	r_3	r_4	r_5
1	3	—	—	—	—
2	M	1	—	—	—
5	M	M	1	0.50	0.33

As before, we need to find the maximal values of M such that

$$\sum_{j=1}^k r_j \cdot \left(\frac{D}{k} + \frac{T}{k} + A \cdot \left(\prod_{i \neq j} \text{Max}\{1, r_i\} - 1 \right) \right) < 3(D + T).$$

We obtain that:

k	2	5
M	4.93	6.22

It follows that

$$\begin{aligned}
 \mathbb{E}(\text{TTM}(2)) &= \frac{4.93}{5.93} \cdot \left(\frac{70}{2} + \frac{180}{2} + 2 \cdot (1 - 1) \right) \\
 &\quad + \frac{1}{5.93} \cdot \left(\frac{70}{2} + \frac{180}{2} + 2 \cdot (4.93 - 1) \right) \approx 126 \text{ days}, \\
 \mathbb{E}(\text{TTM}(5)) &= \frac{6.22 + 6.22}{14.27} \cdot \left(\frac{70}{5} + \frac{180}{5} + 2 \cdot (1 \cdot 1 \cdot 1 \cdot 6.22 - 1) \right) \\
 &\quad + \frac{1 + \frac{1}{2} + \frac{1}{3}}{14.27} \cdot \left(\frac{70}{5} + \frac{180}{5} + 2 \cdot (1 \cdot 1 \cdot 6.22 \cdot 6.22 - 1) \right) \\
 &\approx 69 \text{ days}.
 \end{aligned}$$

One can clearly see that for this example, splitting into five modules gives the best results.

We have shown in this section that splitting into modules while staying within the capacity is possible and can result in a shorter mean time-to-market. However, it is essential for the validity of the results to have reliable information on the distribution of the number of requests over the modules.

3.5 A model including waiting times

In this section we introduce a queueing model to study the mean time-to-market of releases of modules. Assume that the number of modules in which we divide the monolith is equal to k . The model is a closed queueing network with two stations, one consisting of k parallel servers and one consisting of a single server and a request queue (see Figure 3.2).

The first station represents the modules for which no new releases are requested. The second station represents the modules for which a new release is requested. After the release of a new version of module i , the next request for a release of module i occurs after an exponentially distributed time with parameter λ_i , $i = 1, \dots, k$. Here, $1/\lambda_i$ is the mean time until the next request for module i occurs. Typically, the λ_i s are different because not all the modules have the same rate of being requested for a new release since there are modules that are more popular than others. Modules requested for a new release queue up in the request queue until they can be served. The server in the second station represents the group of approximately 400 employees working on the modules. We assume that the server works in a processor sharing fashion. Whenever there are j requests in the request queue the server splits its capacity equally over the j requests. The service time of module i in the second station is exponentially distributed with parameter μ_i , $i = 1, \dots, k$. Here $1/\mu_i$ is the

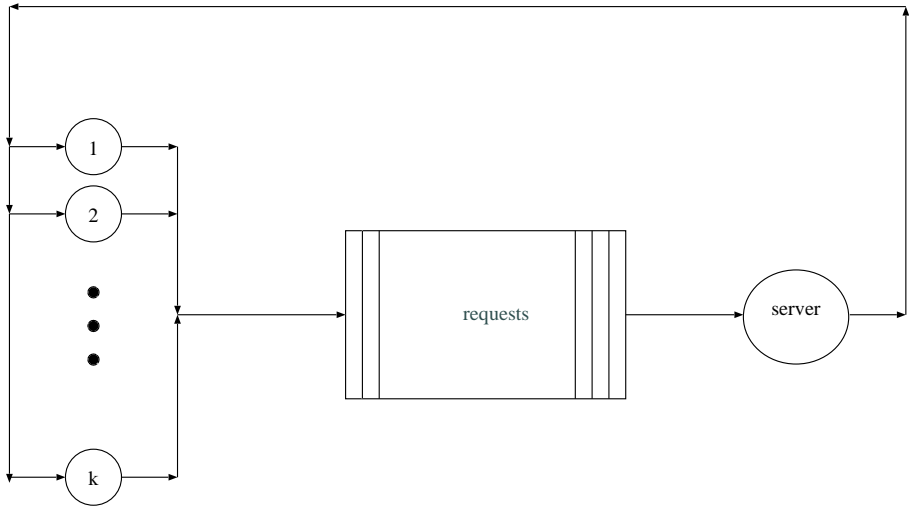


Figure 3.2: The closed queueing network

mean time a request would spend in the second station whenever there would be no other requests at the same time at this station.

The random variable $X_t^{(i)}$, $i = 1, \dots, k$, denotes the state of module i at time t , i.e.,

$$X_t^{(i)} = \begin{cases} 0 & \text{if a new release for module } i \text{ is requested at time } t, \\ 1 & \text{otherwise.} \end{cases}$$

The stochastic process $X(t) = (X_t^{(1)}, X_t^{(2)}, \dots, X_t^{(k)})$ is a continuous-time Markov process. The equilibrium distribution of this continuous-time Markov process can be obtained by solving the balance equations, equating the inflow and outflow of each state, together with the normalization equation. This is illustrated by the following example.

Example 3.5.1. *Let us consider the case $k = 3$. We write p_{i_1, i_2, i_3} , where $i_1, i_2, i_3 \in \{0, 1\}$, to denote the equilibrium probability of the system to be in the state (i_1, i_2, i_3) . For example $p_{0,0,0}$ is the probability that new releases for all modules are requested and $p_{1,1,1}$ is the probability that no new releases are requested. The balance equations are given by*

$$\begin{aligned}
\left(\frac{\mu_1}{3} + \frac{\mu_2}{3} + \frac{\mu_3}{3}\right) p_{0,0,0} &= \lambda_1 p_{1,0,0} + \lambda_2 p_{0,1,0} + \lambda_3 p_{0,0,1}, \\
\left(\frac{\mu_2}{2} + \lambda_1 + \frac{\mu_3}{2}\right) p_{1,0,0} &= \frac{\mu_1}{3} p_{0,0,0} + \lambda_2 p_{1,1,0} + \lambda_3 p_{1,0,1}, \\
\left(\frac{\mu_1}{2} + \lambda_2 + \frac{\mu_3}{2}\right) p_{0,1,0} &= \frac{\mu_2}{3} p_{0,0,0} + \lambda_1 p_{1,1,0} + \lambda_3 p_{1,0,1}, \\
\left(\frac{\mu_1}{2} + \lambda_3 + \frac{\mu_2}{2}\right) p_{0,0,1} &= \frac{\mu_3}{2} p_{0,0,0} + \lambda_1 p_{1,0,1} + \lambda_3 p_{0,1,1}, \\
(\mu_3 + \lambda_1 + \lambda_2) p_{1,1,0} &= \frac{\mu_2}{2} p_{1,0,0} + \frac{\mu_1}{2} p_{0,1,0} + \lambda_3 p_{1,1,1}, \\
(\mu_2 + \lambda_1 + \lambda_3) p_{1,0,1} &= \frac{\mu_3}{2} p_{0,1,0} + \frac{\mu_1}{2} p_{0,0,1} + \lambda_2 p_{1,1,1}, \\
(\mu_1 + \lambda_2 + \lambda_3) p_{1,1,1} &= \frac{\mu_3}{2} p_{0,1,0} + \frac{\mu_2}{2} p_{0,0,1} + \lambda_1 p_{1,1,1}, \\
(\lambda_1 + \lambda_2 + \lambda_3) p_{1,1,1} &= \mu_3 p_{1,1,0} + \mu_2 p_{1,0,1} + \mu_1 p_{0,1,1},
\end{aligned}$$

and the normalization equation is

$$p_{0,0,0} + p_{1,0,0} + p_{0,1,0} + p_{0,0,1} + p_{1,1,0} + p_{1,0,1} + p_{0,1,1} + p_{1,1,1} = 1.$$

Solving the above system of equations we obtain

$$\begin{aligned}
p_{0,0,0} &= 6C\lambda_1\lambda_2\lambda_3, \quad p_{1,0,0} = 2C\lambda_2\lambda_3\mu_1, \quad p_{0,1,0} = 2C\lambda_1\lambda_3\mu_2, \\
p_{0,0,1} &= 2C\lambda_1\lambda_2\mu_3, \quad p_{0,1,1} = C\lambda_1\mu_2\mu_3, \quad p_{1,1,0} = C\lambda_3\mu_1\mu_2, \\
p_{1,0,1} &= C\lambda_2\mu_1\mu_3, \quad p_{1,1,1} = C\mu_1\mu_2\mu_3.
\end{aligned}$$

where

$$\begin{aligned}
\frac{1}{C} &= 6\lambda_1\lambda_2\lambda_3 + 2\lambda_2\lambda_3\mu_1 + 2\lambda_1\lambda_3\mu_2 + 2\lambda_1\lambda_2\mu_3 \\
&\quad + \lambda_3\mu_1\mu_2 + \lambda_2\mu_1\mu_3 + \lambda_1\mu_2\mu_3 + \mu_1\mu_2\mu_3.
\end{aligned}$$

In the case of an arbitrary number of modules we can also obtain a closed expression for the equilibrium distribution, see [1]. The equilibrium probabilities are given by

$$p_{i_1, i_2, \dots, i_k} = C \cdot \left(k - \sum_{j=1}^k i_j\right)! \cdot \prod_{j=1}^k \left(\lambda_j^{1-i_j} \cdot \mu_j^{i_j}\right).$$

where C is chosen such that the sum of the probabilities equals one.

Once we know the equilibrium distribution, we can obtain other performance measures for the system. We denote by $\mathbb{E}(L(k))$ the mean number of modules in the request queue and by $\mathbb{E}(\text{TTM}(k))$ the mean time-to-market for an arbitrary module, i.e., the time between the instant that the release is requested and the instant the new version of the module is released. We can

easily relate these two measures using Little's formula, see e.g. [10]. If $\delta(k)$ is the rate at which new releases for modules are requested, Little's formula gives $\mathbb{E}(L(k)) = \delta(k)\mathbb{E}(\text{TTM}(k))$. The mean number of modules in the request queue and the rate at which new releases for modules are requested can be calculated using

$$\begin{aligned}\mathbb{E}(L(k)) &= \sum_{i \in I} p_{i_1, i_2, \dots, i_k} \cdot \left(k - \sum_{j=1}^k i_j \right), \\ \delta(k) &= \sum_{i \in I} p_{i_1, i_2, \dots, i_k} \cdot \left(\sum_{j=1}^k (i_j \cdot \lambda_j) \right)\end{aligned}$$

with

$$I = \{i = (i_1, \dots, i_k) : i_j \in \{0, 1\} \text{ for all } j\}.$$

The mean time-to-market finally follows from Little's formula.

In the model described in this section, for each module only one request for a new release can be in the request queue. If two or more requests for a module can be simultaneously in the request queue, the model should be adapted. When there can be at most a fixed number of requests for a module simultaneously in the request queue, this can be included in the model by increasing the number of modules in the closed queueing network (e.g. from k to $3k$ if there are at most 3 requests for a module simultaneously in the request queue). If the number of requests simultaneously in the request queue for a certain module is unlimited, probably an open queueing model instead of a closed queueing model is more appropriate. For these open models also results are available for the mean time jobs spend in the system (see e.g. [11] for a formula for the mean time a job spends in the system in an $M/G/1$ processor sharing queue).

3.6 Pairwise Testing

In this section we introduce a method for reducing the number of test cases and therefore for reducing the test effort. We are not interested in functional unit testing but in cross testing, i.e., testing different versions of modules against each other. The testing effort depends on the number of versions of the other modules because a new version of a module should be tested with all combinations of all versions of the other modules. This testing method is known as exhaustive testing. This way of testing covers all test cases. Due to its high cost, to accomplish exhaustive testing in practice is in most cases not feasible. In contrast to exhaustive testing, pairwise testing is only covering all pairwise combinations of versions of modules. This means that for any two modules M_1 and M_2 and any two versions V_1 of M_1 and V_2 of M_2 , there is a test in which M_1 has version V_1 and M_2 has version V_2 .

Different test generation strategies have been published for pairwise testing. Here we briefly describe three of them. In the first approach, if all the pairs in a given combination exist in other combinations we drop that combination, see [3]. Table 3.1 shows the test cases if we consider to divide our software into three modules and to support two versions. In practice we should drop the test cases number two and number four since pairwise they exist already. The second case exists in the test cases 3, 5 and 6. The fourth case exists in the test cases 1, 5 and 6.

Test Cases	Module 1	Module 2	Module 3
1	Version 1	Version 2	Version 2
2	Version 2	Version 1	Version 1
3	Version 2	Version 1	Version 2
4	Version 1	Version 2	Version 1
5	Version 2	Version 2	Version 1
6	Version 1	Version 1	Version 1

Table 3.1: Test cases for 3 modules and 2 supported versions using pairwise techniques.

A combinatorial design approach is used by the Automatic Efficient Test Generator (AETG). This strategy requires that every pair is covered at least once. It does not specify how many times each pair is covered. For further details, see [5] and [6]. A third approach is to use orthogonal arrays to generate test cases. Orthogonal arrays are combinatorial designs used to design statistical experiments that require that every pair is covered the same number of times, see [8].

There are many tools available for generating test cases based on pairwise testing. Each of them is using some specific algorithm for generating pairs. The examples shown in this section are generated using a free GUI based tool for generating test cases called CTE-XL. This tool generates the pairs using the Classification-Tree Method which is a testing method used by DaimlerChrysler AG. For further details about the tool, see [12].

Suppose we divide the software into 3 modules and we want to support 3 versions for each of them. Exhaustive testing requires 27 test cases to cover all possible combinations. However using pairwise testing techniques only 9 test scenarios are required, see Table 3.2.

In Tables 3.3 and 3.4 we compare the number of test cases produced using pairwise testing with the number of test cases produced using exhaustive testing. We consider different number of modules and different number of old supported versions. Table 3.3 shows the number of test cases needed using pairwise testing for 2, 3, 4 and 5 modules supporting 2, 3 and 4 old versions respectively. The number of test cases needed using exhaustive testing for 2, 3, 4 and 5 modules supporting 2, 3 and 4 old versions respectively are shown

Test Cases	Module 1	Module 2	Module 3
1	Version 3	Version 2	Version 3
2	Version 1	Version 3	Version 2
3	Version 2	Version 1	Version 1
4	Version 1	Version 1	Version 3
5	Version 2	Version 2	Version 2
6	Version 3	Version 3	Version 1
7	Version 1	Version 2	Version 1
8	Version 2	Version 3	Version 3
9	Version 3	Version 1	Version 2

Table 3.2: Test cases for 3 modules and 3 supported versions using pairwise techniques.

in Table 3.4. Clearly, the number of test cases increases with the number of modules and with the number of supported versions. Furthermore, we see that if we split up the monolith, for example, into four modules supporting four versions the number of test cases using exhaustive testing grows much faster (256) than using pairwise testing (20).

	2 Modules	3 Modules	4 Modules	5 Modules
2 Versions	4	4	5	6
3 Versions	9	9	9	13
4 Versions	16	19	20	23

Table 3.3: Number of test cases using pairwise testing

	2 Modules	3 Modules	4 Modules	5 Modules
2 Versions	4	8	16	32
3 Versions	9	27	81	256
4 Versions	16	64	256	1024

Table 3.4: Number of test cases using exhaustive testing

Of course, it is possible that pairwise testing alone does not detect all bugs. Sometimes they can be found out only by inspecting three or more module interactions. The possible solution could be to complement pairwise testing with another kind of testing or to extend it to all 3-module (or n -module) combinations, but this could also be costly.

Finally let us remark that in the examples we have presented in this section we have assumed for simplicity that we support the same number of versions for each module. In practice this assumption is not always true since we can support different number of versions for every module. In this case we will have some repeated pairs. Another approach, however, could be to use orthogonal arrays to generate the test cases in which all the pairs are covered the same number of times. For further details and applications, see [9].

3.7 Conclusions

We have translated the problem given to us by ASML into a general model that can be extended to include capacity constraint or waiting times. A lot of parameters appear in this model for which a suitable value should be chosen. One of the most important parameters relates to the popularity rate of the different modules which needs to be investigated by ASML to draw the right conclusions. To illustrate the model we worked out a few examples. For this examples it seems that splitting the monolith into a small number of modules can certainly be an improvement. Other techniques, such as pairwise testing, can be used to further reduce the testing time. We should note that once the optimal number of partitions is derived, a lot of work remains to be done. Actually splitting the software into k more or less independent pieces can be very hard. It may be desirable to deviate from the optimal value to make space for natural partitions (such as splitting firmware and user interface).

Acknowledgement. The authors greatly acknowledge stimulating discussions with Tammo van den Berg, Martijn van Noordenburg and Joost Smits from ASML, and with Nebojsa Gvozdenovic (CWI), Małwina Luczak (London School of Economics) and Rob van der Mei (CWI and Vrije Univeriteit Amsterdam).

3.8 Bibliography

- [1] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *J. ACM*, 22(2):248–260, 1975.
- [2] R. Bisseling et al. Partitioning a call graph. In *Proceedings of the Fifty-second European Study Group with Industry, CWI syllabus 55*, pages 95–107, 2006.
- [3] M. Bolton. Pairwise testing. <http://www.developsense.com/testing/PairwiseTesting.html>, 2004.
- [4] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and zipf-like distributions: evidence and implications. In *Proceedings of IEEE Infocom'99*, pages 126–134, New York, March 1999.

- [5] D. M. Cohen, S. R. Dalal, M. L. Fredman, and G. C. Patton. The AETG system: An approach to testing based on combinatorial design. *IEEE Transactions on Software Engineering*, 23(7):437–444, 1997.
- [6] D. M. Cohen, S. R. Dalal, J. Parelius, and G. C. Patton. The combinatorial design approach to automatic test generation. *IEEE Software*, pages 83–88, 1996.
- [7] C. A. Cunha, A. Bestavros, and M. E. Crovella. Characteristics of WWW client-based traces. Technical Report TR-95-010, Boston University Department of Computer Science, April 1995. Revised July 18, 1995.
- [8] E. Dustin. Orthogonally speaking. *STQE magazine*, 3(5):46–51, October 2001.
- [9] A. S. Hedayat, N. J. A. Sloane, and J. Stufken. *Orthogonal Arrays: Theory and Applications*. Springer-Verlag, New York, 1999.
- [10] L. Kleinrock. *Queueing Systems, Volume I: Theory*. Wiley Interscience, New York, 1975.
- [11] L. Kleinrock. *Queueing Systems, Volume II: Computer Applications*. Wiley Interscience, New York, 1976.
- [12] E. Lehmann and J. Wegener. Test case design by means of the CTE XL. In *Proceedings of the 8th European International Conference on Software Testing, Analysis and Review. EuroSTAR 2000*, Copenhagen, Denmark, December 2000.
- [13] G. K. Zipf. Relative frequency as a determinant of phonetic change. *Harvard Studies in Classical Philology*, 40:1–95, 1929.

CATCHING GAS WITH DROPLETS

Modelling and simulation of a diffusion-reaction process

Simon van Mourik¹, Yves van Gennip², Mark Peletier²,
Andriy Hlod², Vadim Shcherbakov³, Peter in 't Panhuis²,
Erwin Vondenhoff², Pieter Eendebak⁴, Jan Bouwe van den Berg⁵

Abstract

The packaging industry wants to produce a foil for food packaging purposes, which is transparent and lets very little oxygen pass. To accomplish this they add a scavenger material to the foil which reacts with the oxygen that diffuses through the foil. We model this process by a system of partial differential equations: a reaction-diffusion equation for the oxygen concentration and a reaction equation for the scavenger concentration. A probabilistic background of this model is given and different methods are used to get information from the model. Homogenization theory is used to describe the influence of the shape of the scavenger droplets on the oxygen flux, an argument using the Fourier number of the foil leads to insight into the dependency on the position of the scavenger and a method via conformal mappings is proposed to find out more about the role of the size of the droplet. Also simulations with *Mathematica* were done, leading to comparisons between different placements and shapes of the scavenger material in one- and two-dimensional foils.

KEYWORDS: pde modeling, chemical reaction, simulation, homogenization, conformal mapping, Fourier number

4.1 Introduction

In the food packaging industry people are interested in developing materials that can shield food from certain gasses, like oxygen. If too much oxygen comes into contact with the food, the rotting process will set in. For a lot of food the tolerable oxygen concentration is in the order of ten parts per million, as listed in figure 4.1. An additional demand on the packaging foil is transparency, since

1: Universiteit Twente, 2: Technische Universiteit Eindhoven, 3: CWI, 4: Universiteit Utrecht, 5: Vrije Universiteit Amsterdam

customers like to see the food before they buy it. As a solution satisfying both demands DSM considers a polymer sheet which contains droplets of a material that reacts with oxygen, the so called scavenger material. Through reaction with this material the concentration of oxygen in the foil decreases and the flux of oxygen through the packaging material is less than it would be in the absence of scavenger droplets.

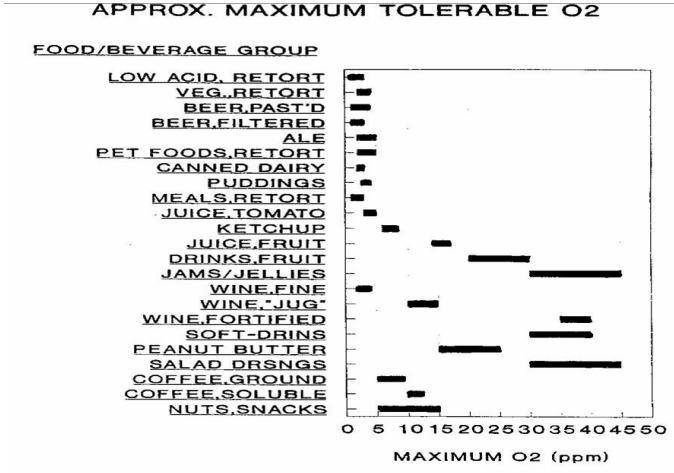


Figure 4.1: Tolerable oxygen concentrations for different types of food.

The question posed by DSM to the *55th European Study Group Mathematics With Industry* was twofold. Firstly they asked the participants to model the diffusion of oxygen gas through a foil containing scavenger droplets. Since DSM is mainly interested in the flux of oxygen through the foil and not so much in the specifics of the diffusion inside the foil, the hope was that the Study Group would come up with a model of the full three-dimensional process, rooted in physics, which could then be simplified to a one-dimensional effective model describing the oxygen flux across the foil. One difficulty in describing the process at hand is the fact that due to reaction with oxygen the scavenger concentration decreases with time.

Secondly DSM was interested in predictions of this model about the influence of the concentration, shape and size of the scavenger droplets on the oxygen flux. The production process is such that after the droplets are added to the packaging material, the foil can be stretched in one of two ways, either uniaxial or biaxial. In the former case the stretching is done in one direction in the plane of the foil, which results in cigar-shaped droplets, in the latter case the stretching takes place in two perpendicular directions in the plane of the foil, resulting in pancake-shaped droplets. In the absence of stretching the droplets remain spherical in first approximation. Furthermore, during addition of the scavenger material, the amount of added scavenger can be controlled as can the size of the added droplets, but the spatial placement of droplets in the

foil cannot. Thus a homogeneous spread of droplets in the material should be assumed.

The problem owners from DSM already came up with a one-dimensional model of the process themselves, which they hoped could be either validated or improved upon during the Study Group. This model consists of a reaction-diffusion equation for the oxygen concentration $\tilde{c}(x, t)$ coupled to a reaction equation for the concentration of scavenger material $\tilde{s}(x, t)$:

$$\frac{\partial \tilde{c}(x, t)}{\partial t} = D \frac{\partial^2 \tilde{c}(x, t)}{\partial x^2} - \kappa_c \tilde{s}(x, t)^\alpha \tilde{c}(x, t)^\beta, \quad (4.1a)$$

$$\frac{\partial \tilde{s}(x, t)}{\partial t} = -\kappa_s \tilde{s}(x, t)^\alpha \tilde{c}(x, t)^\beta, \quad (4.1b)$$

where D is the diffusion coefficient for oxygen in the polymer foil, κ_c is the reaction rate of oxygen and κ_s is the reaction rate of scavenger material. All these coefficients as well as α and β are taken to be constant in space and time. The values of these constants are fully determined by the properties of the foil, the scavenger material and oxygen and are not dependent on the placement, size or shape of the scavenger droplets. Multiplying equation (4.1a) by κ_s and equation (4.1b) by κ_c and subsequently re-scaling the oxygen and scavenger concentrations as $c(x, t) := \kappa_s \tilde{c}(x, t)$ and $s(x, t) := \kappa_c \tilde{s}(x, t)$ leads to the following system of partial differential equations, where now the reaction constant $\kappa := \kappa_c^{-\alpha+1} \kappa_s^{-\beta+1}$ is the same in both equations:

$$\frac{\partial c(x, t)}{\partial t} = D \frac{\partial^2 c(x, t)}{\partial x^2} - \kappa s(x, t)^\alpha c(x, t)^\beta, \quad (4.2a)$$

$$\frac{\partial s(x, t)}{\partial t} = -\kappa s(x, t)^\alpha c(x, t)^\beta. \quad (4.2b)$$

According to DSM, experimental results indicate that $\alpha \approx \frac{5}{3}$ and $\beta \approx 1$. However in our treatment of this model we will often take both these constants to be 1 for simplicity. The system of equations (4.2a) and (4.2b) will be called the *DSM model* from here on.

Both questions DSM asked the Study Group sparked a lot of different initiatives which led to some useful insights into the problem. In this report the different approaches to the proposed problems are discussed and practical results useful for DSM as well as possible new directions for research will be given. The setup for this report will be as follows. In section 4.2 a probabilistic model for the physics on the micro scale is given and the relation between this model and the DSM model of partial differential equations is discussed. Section 4.3 applies the theory of homogenization to a three dimensional generalization of the DSM model with constant scavenger concentration. This approach results in, among other things, a limit problem in which the influence of the shape of the droplet is felt via the first eigenvalue of the Laplacian on a cube with a droplet shaped cavity. In section 4.4 the effects of the position of the scavenger are investigated analytically. The final analytical approach which was undertaken comprises the use of conformal mappings to transform the stationary

problem, i.e. the Laplace equation, on an infinite strip with a rectangular hole to a similar problem on the complex half plane and can be found in section 4.5. The strip with the rectangular hole models the polymer foil with a rectangular droplet of scavenger material in it. The last of the approaches proposed during the Study Group week that will be discussed in this report is the numerical one and can be found in sections 4.6 and 4.7. Simulations in *Mathematica* of the DSM model in one and two space dimensions were made, leading to some new insights in the effects of placement and shape of the scavenger material on the oxygen flux through the foil. In the final section 4.8 the results and conclusion that we think are of greatest interest to DSM, will be restated.

4.2 Probabilistic approach

In this section we propose a possible idealized stochastic microscopic model of catching oxygen by droplets. The model is based on the following three assumptions regarding the chemistry and the physics of the phenomena.

- **A1:** Oxygen molecules move *independently*. The free individual dynamics of any oxygen molecule is simple diffusion that has no preferential directions.
- **A2:** Oxygen molecules interact with droplets *independently* of each other.
- **A3:** When a particle (molecule) hits a droplet, then there is a chance that it can be annihilated. If the reaction takes place, then it affects the droplet too. Namely, it makes the droplet's catching properties worse. Therefore the effectiveness of the reaction process decreases in time.

As operator space, we take the lattice \mathbf{Z}^d , where \mathbf{Z} is a set of integers and d can be 1, 2 or 3. For definiteness we assume $d = 1$ in the sequel. Assumption **A1** suggests that we can model the oxygen molecules by independent simple symmetric random walks $\{x_i(t) \in \mathbf{Z}, t \geq 0, i = 1, 2, \dots\}$. Let us explain informally what a simple random walk on the lattice is. One can think of a particle that moves on \mathbf{Z} as follows. Assume that a particle is at point $k \in \mathbf{Z}$ at time $t \geq 0$. The particle sits at this point for a random time t_k (which is an exponentially distributed random variable with parameter $\lambda > 0$), then it jumps to either site $k - 1$ or $k + 1$ chosen with probability 1/2. It occupies the new location for another exponentially distributed random time (which is independent of t_k), and jumps again to one of the nearest neighbors chosen equally likely and so on. Formally, a simple symmetric random walk $x(t)$, $t \geq 0$, on the lattice \mathbf{Z} is a continuous time Markov chain whose dynamic is specified by the following infinitesimal probabilities

$$\mathbf{P} \{x(t + \delta t) = y | x(t) = x\} = \begin{cases} D\delta t/2 + O(\delta t), & \text{if } |x - y| = 1, \\ 1 - D\delta t + O(\delta t), & \text{if } x = y, \\ 0, & \text{if } |x - y| > 1, \end{cases}$$

where $x, y \in \mathbf{Z}$, D is a diffusion coefficient and $O(\delta t) \rightarrow 0$ as $\delta t \rightarrow 0$. D specifies the intensity of jumps and it is equal to λ^{-1} .

The random walks interact with a random medium modelling scavenger droplets. In the DSM model the distribution of the scavenger material is specified by its concentration at every point inside the foil. In our model we introduce a system of independent nonnegative integer valued random processes $\{\eta_k(t), k \in \mathbf{Z}, t \geq 0\}$ which are interpreted as amounts of scavenger material at the lattice points at time t .

Let us describe an interaction between particles and droplets. Denote by $\xi_k(t)$ a number of particles at point $k \in \mathbf{Z}$ at time t . Consider a point k such that $\xi_k(t) > 0$ and $\eta_k(t) > 0$. Within the time interval $[t, t + \delta t)$ any oxygen molecule at this point can react with a droplet with probability $F(\eta_k(t))\delta t + O(\delta t)$, where $O(\delta t) \rightarrow 0$ as $\delta t \rightarrow 0$. Here $F(\cdot)$ is some nonnegative function, such that $F(0) = 0$. With probability $1 - F(\eta_k(t))\delta t + O(\delta t)$ the molecule does not react. As a result of the reaction the oxygen molecule and a certain amount of scavenger material annihilate each other, therefore $\xi_k(t) \rightarrow \xi_k(t) - 1$ and $\eta_k(t) \rightarrow \eta_k(t) - 1$ respectively. When all the scavenger material is reacted away, then subsequently the oxygen molecules diffuse passively through without being affected.

Let us compute the probability of the event that at least one of the oxygen molecules reacts at point k during the time interval $[t, t + \delta t)$. Assumption **A2** yields that the probability of the event that exactly $0 < j < m = \min(\xi_k(t), \eta_k(t))$ molecules react during time interval $[t, \delta t)$ is given by

$$\binom{m}{j} (F(\eta_k(t))\delta t + O(\delta t))^j (1 - F(\eta_k(t))\delta t + O(\delta t))^{m-j}$$

and it is negligible in comparison with $F(\eta_k(t))\delta t$ as δt goes to 0. Therefore the probability of the event that at least one molecule reacts during the time interval $[t, t + \delta t)$ is equal to

$$\xi_k(t)F(\eta_k(t))\delta t + O(\delta t), \quad (4.3)$$

and it is the probability of the event that exactly one of the molecules reacts within the same time interval. The coefficient in front of δt , i.e. $\xi_k(t)F(\eta_k(t))$ is, by definition, the total reaction rate at point k at time t .

Back to the continuum equations

In this section we show the connection between the microscopic probabilistic model and the following system of partial differential equations

$$\frac{\partial c(x, t)}{\partial t} = D \frac{\partial^2 c(x, t)}{\partial x^2} - F(s(x, t))c(x, t), \quad (4.4a)$$

$$\frac{\partial s(x, t)}{\partial t} = -F(s(x, t))c(x, t). \quad (4.4b)$$

In particular, if $F(x) = \kappa x^{5/3}$, then we get the DSM model. We would like to highlight the main idea and will not go into many technical details. The idea is to consider the stochastic system at points $k(\varepsilon) \sim [x/\varepsilon]$, where $x \in \mathbf{R}$, after time $t(\varepsilon) = t/\varepsilon^2$ and then pass to the limit $\varepsilon \rightarrow 0$. It is a so-called hydrodynamic limit in the standard terminology of statistical physics. Points $k(\varepsilon)$ are microscopic points, $t(\varepsilon)$ is microscopic time. Respectively, continuous point x is called a macroscopic one and t is the macroscopic time. The reaction rates should be rescaled respectively, since the impact of the reaction at any microscopic point should be negligible at macroscopic scale, but the effect of the reaction is visible in a continuum domain with positive volume. Namely, we put the reaction rate equal to $\varepsilon^2 F(\cdot)$ (in a few lines this choice will become clear). Formally, the pair of random processes $(\xi(t/\varepsilon^2), \eta(t/\varepsilon^2))$ forms a Markov process with the state space $S = \{(\xi, \eta) \in (\mathbf{Z}_+ \cup \{0\})^\infty \times (\mathbf{Z}_+ \cup \{0\})^\infty\}$ and with the following infinitesimal operator

$$\begin{aligned} G_\varepsilon f(\xi, \eta) = & \varepsilon^{-2} D \sum_k (f(\xi + e_{k+1} - e_k, \eta) - f(\xi, \eta)) \xi_k \\ & + \varepsilon^{-2} D \sum_k (f(\xi + e_{k-1} - e_k, \eta) - f(\xi, \eta)) \xi_k \\ & + \varepsilon^{-2} \sum_k (f(\xi - e_k, \eta - e_k) - f(\xi, \eta)) \xi_k F(\eta_k) \varepsilon^2 \end{aligned} \quad (4.5)$$

where $e_k \in \mathbf{Z}^\infty$ are infinite dimensional vectors with all zero components except the k th, which are equal to 1. Existence of this process can be proved by the general methods of the theory of interacting particle systems, we refer to [5] for more details. The factor ε^{-2} in front of the sums manifests the fact that we speed up the process time. Obviously, in the third sum it cancels out with its reciprocal in the reaction term $\xi_k F(\eta_k) \varepsilon^2$. For any $\varepsilon > 0$ consider a random process

$$J_\varphi^{(\varepsilon)}(t) = \varepsilon \sum_{k \in \mathbf{Z}} \varphi(\varepsilon k) \xi_k(t/\varepsilon^2),$$

where $\varphi : \mathbf{R} \rightarrow \mathbf{R}$ is an integrable bounded function (test function). $J_\varphi^{(\varepsilon)}(t)$ is nothing else but an integral of the function φ with respect to the measure

$$\mu_t^\varepsilon = \varepsilon \sum_{k \in \mathbf{Z}} \xi_k(t/\varepsilon^2) \delta_{\{\varepsilon k\}}(\cdot), \quad (4.6)$$

where $\delta_{\{x\}}(\cdot)$ is a delta function at point $x \in \mathbf{R}$. It follows from the theory of Markov processes that we can write

$$J_\varphi^{(\varepsilon)}(t) = J_\varphi^{(\varepsilon)}(0) + \int_0^t G_\varepsilon J_\varphi^{(\varepsilon)}(s) ds. \quad (4.7)$$

Direct computations show that

$$\begin{aligned} G_\varepsilon J_\varphi^{(\varepsilon)}(s) &= \varepsilon^{-1} D \sum_{k \in \mathbf{Z}} (\varphi(k\varepsilon + \varepsilon) - \varphi(k\varepsilon)) \xi_k(s/\varepsilon^2) \\ &\quad + \varepsilon^{-1} D \sum_{k \in \mathbf{Z}} (\varphi(k\varepsilon - \varepsilon) - \varphi(k\varepsilon)) \xi_k(s/\varepsilon^2) \\ &\quad + \sum_k (\varphi(k\varepsilon - \varepsilon) - \varphi(k\varepsilon)) \xi_k(s/\varepsilon^2) F(\eta_k(s/\varepsilon^2)). \end{aligned}$$

Assuming that function φ is smooth enough we can use the Taylor expansion and obtain that

$$\begin{aligned} G_\varepsilon J_\varphi^{(\varepsilon)}(s) &= \varepsilon D \sum_{k \in \mathbf{Z}} \varphi''(k\varepsilon) \xi_k(s/\varepsilon^2) - \varepsilon \sum_{k \in \mathbf{Z}} \varphi'(k\varepsilon) \xi_k(s/\varepsilon^2) F(\eta_k(s/\varepsilon^2)) \\ &\quad + R_\varphi(\varepsilon), \end{aligned}$$

where $R_\varphi(\varepsilon) \rightarrow 0$ in probability as $\varepsilon \rightarrow 0$. Substituting it into the equation (4.7) we obtain

$$\begin{aligned} \varepsilon \sum_{k \in \mathbf{Z}} \varphi(\varepsilon k) \xi_k(t/\varepsilon^2) - \varepsilon \sum_{k \in \mathbf{Z}} \varphi(\varepsilon k) \xi_k(0) &= \varepsilon D \int_0^t \sum_{k \in \mathbf{Z}} \varphi''(k\varepsilon) \xi_k(s/\varepsilon^2) ds \quad (4.8) \\ &\quad - \varepsilon \int_0^t \sum_{k \in \mathbf{Z}} \varphi'(k\varepsilon) \xi_k(s/\varepsilon^2) F(\eta_k(s/\varepsilon^2)) ds + R_\varphi(\varepsilon). \end{aligned}$$

Repeating the same arguments for a random process

$$\varepsilon \sum_{k \in \mathbf{Z}} \psi(\varepsilon k) \eta_k(t/\varepsilon^2)$$

which is an integral of test function ψ with respect to another measure

$$\nu_t^{(\varepsilon)} = \varepsilon \sum_{k \in \mathbf{Z}} \eta_k(t/\varepsilon^2) \delta_{\{\varepsilon k\}}(\cdot),$$

we obtain that

$$\begin{aligned} \varepsilon \sum_{k \in \mathbf{Z}} \psi(\varepsilon k) \eta_k(t/\varepsilon^2) - \varepsilon \sum_{k \in \mathbf{Z}} \psi(\varepsilon k) \eta_k(0) \quad (4.9) \\ = -\varepsilon \int_0^t \sum_{k \in \mathbf{Z}} \psi'(k\varepsilon) \xi_k(s/\varepsilon^2) F(\eta_k(s/\varepsilon^2)) ds + R_\psi(\varepsilon), \end{aligned}$$

where $R_\psi(\varepsilon) \rightarrow 0$ in probability as $\varepsilon \rightarrow 0$. So, informally, we can conclude that the pair $(\xi(t/\varepsilon^2), \eta(t/\varepsilon^2))$ "mimics" a weak solution of the system of equations (4.4a) and (4.4b). To see this, one should replace in equations (4.8) and (4.9)

the random processes $\xi(\tau/\varepsilon^2)$ and $\eta(\tau/\varepsilon^2)$ by functions $c(x, \tau)$ and $s(x, \tau)$ respectively, and the sums over k should be replaced by integrals. If functions $(c(x, \tau)$ and $s(x, \tau))$ satisfy such integral equations for any smooth enough finitely supported functions φ and ψ , then, by definition, they form a weak solution of the system of equations (4.4a) and (4.4b).

The reasoning above can be placed in a rigorous setting of the modern theory of hydrodynamic limits for interacting particle systems [4]. Using the general methods of this theory it is possible to prove that the Markov process $(\xi(t/\varepsilon^2), \eta(t/\varepsilon^2))$, $t \geq 0$, converges in some rigorous sense to a weak solution of the system of equations (4.4a)–(4.4b) as $\varepsilon \rightarrow 0$. It can be shown (using the methods of the theory of partial differential equations) that there exists a unique weak solution in this case. Then, it remains to note that any strong solution is a weak solution. Hence, uniqueness of the strong solution implies that the obtained weak solution is in fact a strong solution of the system of equations (4.4a)–(4.4b).

Experimental results of DSM show that the equation for the oxygen concentration $c(t, x)$ should be linear in $c(t, x)$. We have just shown that our microscopic probabilistic model leads to this type of equations. This is determined by the fact that the total reaction rate (see equation (4.3)) is linear in the number of oxygen molecules at a point due to Assumption **A2**.

We would suggest a simulation study of the proposed stochastic model. In this study the described particle system can be simulated in a finite lattice volume with certain boundary conditions and the droplet shapes can be taken into account. The parameters of the simulated model should be specified in collaboration with DSM in order to have a plausible approximation to the real situation.

4.3 Homogenization

In this section we study the limit of small length size and derive a description in terms of homogenized quantities.

The starting point for our discussion is the system of equations

$$c_t = D\Delta c - kcs \quad \text{for } x \in \Omega, \ t > 0 \quad (4.10a)$$

$$s_t = -kcs \quad \text{for } x \in \Omega, \ t > 0 \quad (4.10b)$$

$$c = c_b \quad \text{for } t > 0 \text{ and } x \in \partial\Omega \quad (4.10c)$$

$$(c, s) = (c_i, s_i) \quad \text{for } t = 0 \text{ and } x \in \Omega. \quad (4.10d)$$

Here Ω is a domain in \mathbb{R}^n representing the foil; c_b is a given boundary value function, c_i and s_i are the initial data for the oxygen and scavenger concentrations, and k and D are reaction and diffusion parameters. The assumption that the scavenger is contained in small inclusions is encoded in the initial datum s_i .

The relevant dimensionless parameter that indicates whether the problem is diffusion- or reaction-dominated is

$$\alpha := \frac{D}{\varepsilon^2 k S}$$

where ε is the typical microscopic length scale (the distance between scavenger inclusions) and S is a typical scale of s . If this number is large, then the diffusion is fast enough to homogenize differences on length scales ε ; if it is small, then the fast reaction creates large local variations of c .

Diffusion-dominated

If α is large, then the concentration c varies little between regions with and without scavenger; one can directly write down the homogenized problem (cf. [2]),

$$c_t = D\Delta c - kcs \quad \text{for } x \in \Omega, \ t > 0 \quad (4.11a)$$

$$s_t = -kcs \quad \text{for } x \in \Omega, \ t > 0 \quad (4.11b)$$

$$c = c_b \quad \text{for } t > 0 \text{ and } x \in \partial\Omega \quad (4.11c)$$

$$(c, s) = (\bar{c}_i, \bar{s}_i) \quad \text{for } t = 0 \text{ and } x \in \Omega, \quad (4.11d)$$

where now \bar{c}_i and \bar{s}_i are locally averaged (macroscopic) concentrations. Note that in this problem the length scale of the scavenger inclusions no longer appears (it is involved indirectly in determining \bar{s}_i).

Reaction-dominated

On the other hand, if α is small, then the reaction forces the concentration of c to zero wherever s is non-zero.

In the limit $\alpha \rightarrow 0$ problem (4.10) converges to a Stefan problem:

$$c_t = D\Delta c \quad \text{in } \{c > 0\} \quad (4.12a)$$

$$c = c_b \quad t > 0, \ x \in \partial\Omega \quad (4.12b)$$

$$v_n = -\frac{1}{s_i} \frac{\partial c}{\partial n} \quad \text{on } \partial\{c > 0\} \setminus \partial\Omega \quad (4.12c)$$

where n is the outward normal to $\{c > 0\}$ and v_n is the velocity of the interface $\partial\{c > 0\}$. Recall that s_i is the initial s -concentration; in the limit the concentration of s at any point x does not change until the interface reaches x . (This convergence result is proved in [1] for one dimension). The domain thus splits in two parts: one where $ks = 0$ and one where $ks \approx \infty$.

As an intermediate problem, we consider the case in which c solves (4.12a) and (4.12b), on a fixed perforated domain, with zero interior boundary condi-

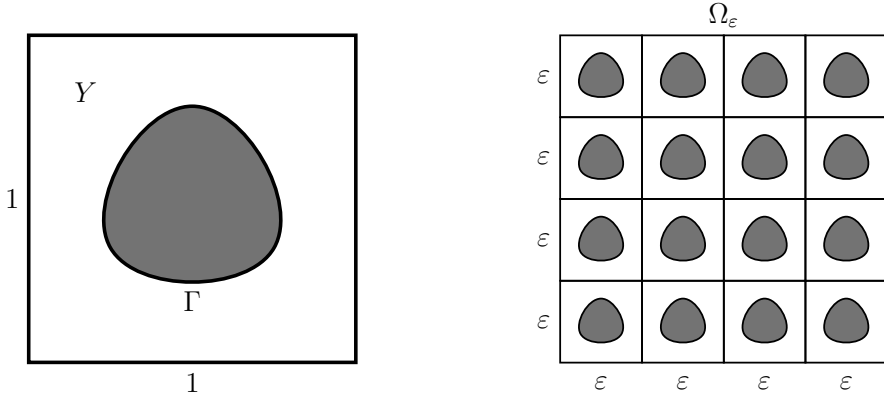


Figure 4.2: Geometry of the unit cell (left) and the perforated domain (right). We have indicated the boundary $\Gamma = Y \cap \partial A$ in the unit periodic cell on the left.

tions:

$$c_t = D\Delta c \quad \text{for } x \in \Omega_\varepsilon, \ t > 0 \quad (4.13a)$$

$$c = 0 \quad \text{for } t > 0 \text{ and } x \in \Gamma_\varepsilon \quad (4.13b)$$

$$c = c_b \quad \text{for } t > 0 \text{ and } x \in \partial\Omega_\varepsilon \setminus \Gamma_\varepsilon \quad (4.13c)$$

$$c = c_i \quad \text{for } t = 0 \text{ and } x \in \Omega_\varepsilon. \quad (4.13d)$$

Here Ω_ε is a perforated version of Ω , constructed in the following way. Let $A \subset \mathbb{R}^n$ be a 1-periodic subset of \mathbb{R}^n (i.e. $A + e_i = A$ for all unit base vectors e_i), and set $A_\varepsilon = \varepsilon A$. We write Y for the unit periodic cell of A , $Y = [0, 1]^n \cap A$. Define Ω_ε by

$$\Omega_\varepsilon = \Omega \cap A_\varepsilon.$$

The internal boundary Γ_ε is given by $\Gamma_\varepsilon = \Omega \cap \partial A_\varepsilon$, see also Figure 4.2.

In the limit $\varepsilon \rightarrow 0$, the ratio of the area of Γ_ε to the volume Ω_ε is unbounded, and therefore the solution of (4.13) converges pointwise to zero as $\varepsilon \rightarrow 0$. We take this fast decay of the solution into account by posing the following *Ansatz* of the solution c^ε :

$$c^\varepsilon(x, t) = e^{-D\lambda t/\varepsilon^2} c_0(x, t) w\left(\frac{x}{\varepsilon}\right),$$

where λ is the first eigenvalue of $-\Delta$ on A with homogeneous Dirichlet boundary conditions. Here c_0 is defined on $\Omega \times [0, \infty)$ and w on Y . On substitution into (4.13a) we find, writing y for x/ε ,

$$\begin{aligned} 0 = & \varepsilon^{-2} c_0(x, t) [D\Delta w(y) + D\lambda w(y)] \\ & + \varepsilon^{-1} 2D\nabla c_0(x, t) \cdot \nabla w(y) \\ & + \varepsilon^0 w(y) [D\Delta c_0(x, t) - c_{0t}(x, t)] \end{aligned} \quad (4.14)$$

By the choice of λ , the equation at level ε^{-2} forces w to be a multiple of the first eigenfunction. This function w is therefore also the first eigenfunction of $-\Delta$ on Y with the following boundary conditions:

$$\begin{aligned} &\text{periodic conditions on } \partial Y \cap \partial[0, 1]^n; \\ &\text{homogeneous Dirichlet conditions on } \partial Y \setminus \partial[0, 1]^n. \end{aligned}$$

Since we can multiply w by a constant, and divide c_0 by the same constant without changing c , we choose to normalize w by assuming

$$\int_Y w(y) dy = 1.$$

Integrating (4.14) over Y and using the periodicity of w , the integral at level ε^{-1} vanishes, and we are left with

$$c_{0t} = D\Delta c_0 \quad \text{for } x \in \Omega, \ t > 0.$$

The function c^ε is therefore approximated by the solution of the equation

$$c_t = D\Delta c - \frac{D\lambda}{\varepsilon^2} c \quad \text{for } x \in \Omega, \ t > 0. \quad (4.15)$$

To make the connection back to the Stefan problem, we note that the parameter λ in (4.15) is determined by solving an eigenvalue problem on the perforated unit cell Y . We now need to make an assumption on how we may deduce the microscopic geometry from a given macroscopic scavenger concentration s —for instance, we could assume that in the unit cell the scavenger is contained in a sphere of concentration \bar{s} ; the macroscopic concentration then determines the radius of this sphere (see also below).

Under such an assumption, the parameter λ is a function of the macroscopic scavenger concentration s , and the macroscopic oxygen concentration c satisfies the equation

$$c_t = D\Delta c - \frac{D\lambda(s)}{\varepsilon^2} c.$$

By mass conservation—the difference $c - s$ is conserved locally in (4.10a), therefore the same is true for the homogenized concentration—the equation for s is

$$s_t = -\frac{D\lambda(s)}{\varepsilon^2} c.$$

By doing this we are treating the geometry as quasi-static in the c -equation, *i.e.* we assume that the geometry does not change on the time scale of the c -equation. This depends on the concentration of s , as can be seen in (4.12c); it means that the ratio C/S is small, where C and S are typical scales of the initial concentrations c_i and s_i .

Finally, let us analyse the asymptotic behaviour of the eigenvalue λ for small scavenger concentrations s . We will approximate the eigenvalue problem for

the Laplacian on a unit cell $Y = [0, 1]^n \cap A$ by one on a unit ball with a much smaller ball $B_{r_0}(0)$ inside. Furthermore, we replace the periodic boundary conditions by Neumann boundary conditions. It is our firm belief that these approximations do not influence the asymptotic result for small s . Since the first eigenfunction will be symmetric, we arrive at the problem

$$\begin{aligned} c_{rr} + \frac{2}{r}c_r + \lambda c &= 0 & \text{for } r_0 < r < 1 \\ c(r_0) &= 0 \\ c_r(1) &= 0 \end{aligned}$$

The general solution of the differential equation is

$$c(r) = C_1 \frac{\sin(\sqrt{\lambda}r)}{r} + C_2 \frac{\cos(\sqrt{\lambda}r)}{r}.$$

Applying the boundary conditions leads, after some calculations, to $\lambda \sim 3r_0$ as $r_0 \rightarrow 0$. In terms of the scavenger concentration s this translates to $\lambda(s) \sim Cs^{1/3}$ for small s .

When we compare this to the analogous two-dimensional problem, representing very elongated droplets, we find $\lambda(s) \sim C|\ln s|^{-1}$ for small s . This suggests that small elongated droplets lead to higher values of λ and thus a more effective scavenger in this limit problem.

Summary

If we assume that the parameter $\alpha = D/\varepsilon^2 kS$ is large, then we find in the limit $\varepsilon \rightarrow 0$ the homogenized equations

$$\begin{aligned} c_t &= D\Delta c - kcs & \text{for } x \in \Omega, \ t > 0 \\ s_t &= -kcs & \text{for } x \in \Omega, \ t > 0. \end{aligned}$$

On the other hand, if α is small, then we find as limit equations

$$c_t = D\Delta c - \frac{D\lambda(s)}{\varepsilon^2}c \quad \text{for } x \in \Omega, \ t > 0 \quad (4.16a)$$

$$s_t = -\frac{D\lambda(s)}{\varepsilon^2}c \quad \text{for } x \in \Omega, \ t > 0 \quad (4.16b)$$

where λ is determined from s as described above, and the asymptotic behaviour for small concentrations is $\lambda \sim Cs^{1/3}$ and $\lambda \sim C|\ln s|^{-1}$ for small spherical and thin elongated droplets, respectively.

There is a paradox in this: the parameter α itself depends on ε . For the diffusion-dominated case this does not matter, since the limit $\varepsilon \rightarrow 0$ is consistent with the assumption that α is small. In the reaction-dominated case, however, the limit $\varepsilon \rightarrow 0$ entails large values of α . The statement above should therefore be understood as a description of intermediate asymptotics: in the parameter regime in which both α and ε are small, the problem (4.16) is expected to approximate the problem (4.10). For any fixed k and D , in the limit $\varepsilon \rightarrow 0$ the system will *eventually* be diffusion-dominated.

4.4 Behavior of the penetration time in one dimension

Although the reaction-diffusion scheme described by equations (4.2a) and (4.2b) is hard to analyze (mainly because of the nonlinearities and the absence of nonzero equilibrium points), it is possible to do so under some physical assumptions, that are explained below. These assumptions were discussed with the problem owners and seemed valid. In section 4.3 the effects of the shape of the droplets is discussed. In this section we consider the one-dimensional case where the packaging material consists of two layers of foil (without scavenger material) and one layer of pure scavenger material (Figure 4.3).

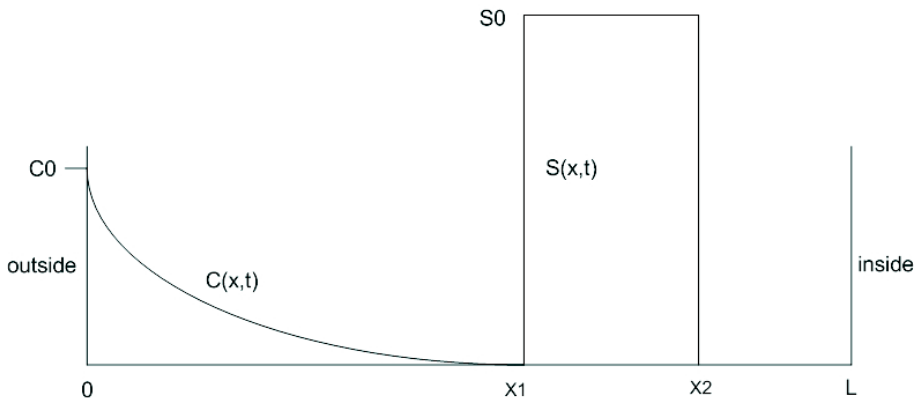


Figure 4.3: One-dimensional cross-section of packaging material

In this figure, C_0 denotes the oxygen concentration on the outside, S_0 the initial scavenger concentration, x_1 and x_2 the initial boundaries of the scavenger layer, $c(x, t)$ the oxygen profile within the packaging material, and L the thickness of the material.

The objective is now to express the time that the oxygen needs to penetrate the material (penetration time) as a function of the physical coefficients. We consider three time intervals; t_1 , t_2 and t_3 , that denote the penetration times for layers 1, 2 and 3 respectively.

Penetration time for a layer of foil

The typical penetration time for the first layer of foil is expressed in terms of the Fourier number for mass transport

$$Fo = \frac{D\tau}{d^2}, \quad (4.17)$$

where D is the mass diffusion coefficient, and τ the penetration time for a layer with thickness d ; see for example [3] or any standard book on physical transport phenomena. This number relates a specifically chosen 'critical' flux

after time τ , to the penetration time for the same flux with different values for parameter d . For some critical flux, Fo can be measured experimentally. Theoretically, we would like to know the time before *some* oxygen reaches the scavenger boundary, but since the process is modelled by a diffusion equation, the oxygen concentration is nonzero over the whole layer of foil instantly. Hence the assumption of a critical flux to mark the penetration time. From equation (4.17) it follows that the penetration time for the first layer of foil is

$$t_1 = \frac{Fox_1^2}{D}. \quad (4.18)$$

Penetration time for a layer of scavenger material

When oxygen reaches x_1 , the scavenger reacts with the oxygen and vanishes. The left side of the layer of scavenger will start reacting away and leave a layer of pure foil. This means that boundary $x_2(t)$ will move to the right. In order to compute the penetration time for a layer of scavenger material, we make two extra assumptions. The first assumption is that S_0 is 'large enough', so that once the oxygen reaches x_1 , $c(x, t)$ settles quickly to an equilibrium profile, while $x_2(t)$ moves only a little. The second assumption is that the scavenger reacts very quickly with oxygen, so that at $x_2(t)$ the oxygen concentration is approximately zero. (The second assumption holds true if the Thiele modulus $Th = d_s \sqrt{\frac{k}{D}}$ is large. This dimensionless quantity indicates the dominance of chemical reaction rate over diffusive mass transfer rate). These assumptions lead us to the following. Once a critical amount of oxygen reaches x_1 , $c(x, t)$ settles quickly into its equilibrium profile, which is a time-varying linear function with boundary conditions $c(0) = C_0$ and $c(x_2(t)) = 0$. According to Fick's law [3] the oxygen flux at $x_2(t)$ is

$$-D \frac{dc(x_2(t))}{dx_2(t)} = \frac{DC_0}{x_2(t)}. \quad (4.19)$$

The amount of scavenger that disappears after reaction is

$$S_0 \frac{dx_2(t)}{dt}. \quad (4.20)$$

Since an amount of β particles of oxygen react with α particles of scavenger, the mass balance reads

$$\beta \frac{DC_0}{x_2(t)} = \alpha S_0 \frac{dx_2(t)}{dt}, \quad (4.21)$$

with initial condition $x(0) = x_1$. The solution to equation (4.21) is

$$x_2(t) = \sqrt{\frac{2\beta C_0 D t}{\alpha S_0}} + x_1^2. \quad (4.22)$$

The penetration time t_2 is obtained by solving

$$x_2 = \sqrt{\frac{2\beta C_0 D t_2}{\alpha S_0}} + x_1^2, \quad (4.23)$$

which gives

$$t_2 = \frac{\alpha S_0(x_1^2 - x_1^2)}{2\beta DC_0} \quad (4.24)$$

$$= \frac{\alpha S_0 d_s (x_1 + x_2)}{2\beta DC_0}, \quad (4.25)$$

with d_s the thickness of the scavenger layer.

Total penetration time

The penetration time for the third layer, t_3 , is computed similarly to t_1

$$t_3 = \frac{\text{Fo}(L - x_2)^2}{D} \quad (4.26)$$

$$= \frac{\text{Fo}(d_s + x_1)^2}{D}. \quad (4.27)$$

The total penetration time is now

$$t_{total} = \frac{\text{Fo}x_1^2}{D} + \frac{\alpha S_0 d_s (x_1 + x_2)}{2\beta DC_0} + \frac{\text{Fo}(d_s + x_1)^2}{D}. \quad (4.28)$$

Equation (4.28) is of the form

$$a_1 x_1 + a_2 S_0 d_s^2 + a_3 S_0 x_1 d_s + a_4 d_s, \quad (4.29)$$

with a_i positive. The total penetration time increases with d_s quadratically, and with x_1 and S_0 linearly.

For further investigation, we would like to pose the idea of relating the penetration time of the scavenger layer to a dimensionless number, similar to the Fourier number. As was mentioned before, the assumptions are discussed with DSM, but of course need scientific validation.

4.5 Solution procedure for the stationary problem via conformal mapping

To test the effect of changing the shape and form of the scavenger droplets inside the foil, a simplified model is assumed, where the foil is modeled as an infinite strip with one block of scavenger in the middle of height $2h$ and width $2L$ (see figure 4.4). The thickness of the strip is rescaled to 2. Obviously we are interested in those values of h and L for which we have the smallest flux of oxygen reaching the food.

The stationary problem

The system of equations (4.2a) and (4.2b) is difficult to solve analytically and does not have any non-trivial stationary solutions. However, a stationary solution can be found if we look at a slightly different problem. We consider

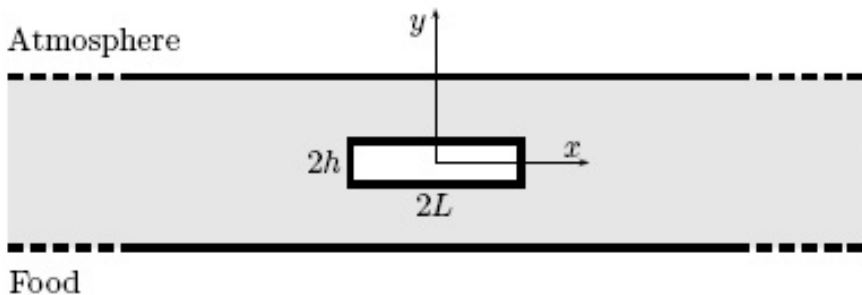


Figure 4.4: Foil modeled as an infinite strip with a rectangular piece of scavenger.

the case of a high reaction speed, so that any oxygen reaching the scavenger boundary will react away immediately. Furthermore the block is assumed to be saturated, so that there is an unlimited supply of scavenger. As a result the oxygen concentration at the boundary of the block will always be equal to zero and the scavenger concentration will be constant in time. Outside the block the oxygen concentration can be described by the diffusion equation, of which the stationary solutions satisfy Laplace's equation. Finally we also assume that the oxygen reacts away with the food immediately so that at the side of the food we also have $c = 0$. At the other side of the foil the concentration of oxygen can be assumed constant and is scaled in such a way that $c = 1$.

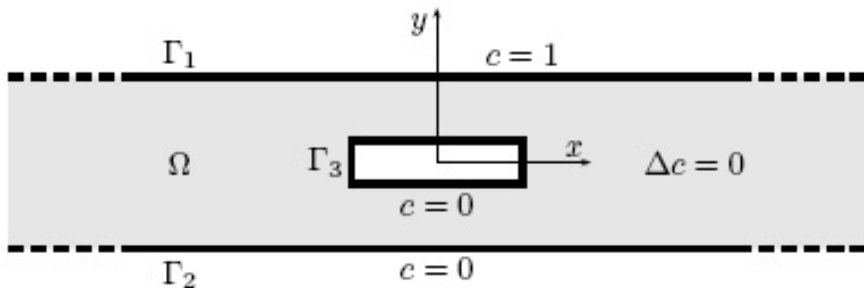


Figure 4.5: Boundary value problem on an infinite strip excluding a rectangular piece of scavenger.

Now let Ω be the two-dimensional domain that consists of an infinite strip excluding a rectangle of length $2L$ and width $2h$:

$$\Omega := \{(x, y) \in \mathbb{R}^2 : -1 < y < 1\} \setminus ([-L, L] \times [-h, h]),$$

and let $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ be the boundary of Ω , where Γ_1 is the boundary with the environment, Γ_2 is the boundary with the food and Γ_3 the boundary of the

block, as indicated in figure 4.5. On this geometry we want to solve

$$\Delta c = 0, \quad (x, y) \in \Omega, \quad (4.30a)$$

subject to

$$c = 1, \quad (x, y) \in \Gamma_1, \quad (4.30b)$$

$$c = 0, \quad (x, y) \in \Gamma_2 \cup \Gamma_3. \quad (4.30c)$$

This problem can describe the oxygen concentration for small time, when none of the scavenger has reacted away completely. Moreover data provided by DSM indicated that initially the amount of permeated oxygen did not change much in time, suggesting stationary behaviour.

The boundary value problem in (4.30) still seems difficult to solve because of the complexity of the domain. However, using the theory of conformal mappings [6], solving Laplace's equation can be reduced to solving a potential problem on an easier domain in the complex plane.

Solution using conformal mappings

We identify the geometry in figure 4.5 with the complex plane. Because of symmetry we can restrict ourselves to that part of Ω where x is positive, which will be denoted by Ω_1 (see figure 4.6):

$$\Omega_1 := \{z \in \mathbb{C} : (\operatorname{Re} z, \operatorname{Im} z) \in \Omega \wedge \operatorname{Re} z > 0\}.$$

Consequently an extra Neumann boundary condition $\partial c / \partial n = 0$ arises at $x = 0$, where n is the outward normal vector. We introduce the points A, B, C, D, E, F and G by $A = i$, $B = hi$, $C = L + hi$, $D = L - hi$, $E = -hi$, $F = -i$ and $G = \infty \pm i$, as indicated in figure 4.6.

A map $f : \Omega_1 \rightarrow \Omega_2$ is called a conformal mapping if $f(z)$ is analytical and one-to-one in Ω_2 . Furthermore $f^{-1} : \Omega_2 \rightarrow \Omega_1$ exists and is also a conformal mapping. Throughout this section we will implicitly use the following two theorems [6]:

Theorem 4.5.1. *Riemann mapping theorem:*

For any two simply connected open subsets Ω_z, Ω_w of the complex plane \mathbb{C} that are not all of \mathbb{C} and for given $z_0 \in \Omega_z, w_0 \in \Omega_w, \alpha \in \mathbb{R}$, there exists a unique conformal map $f : \Omega_z \rightarrow \Omega_w$ such that $f(z_0) = w_0$ and $\arg(f'(z_0)) = \alpha$.

Theorem 4.5.2. *Carathéodory's theorem:*

For any pair of simply connected open sets Ω_z and Ω_w bounded by Jordan curves Γ_z and Γ_w , a conformal map $f : \Omega_z \rightarrow \Omega_w$ can be extended continuously to the boundary, giving a homeomorphism $F : \Gamma_z \rightarrow \Gamma_w$. Furthermore, if z follows the boundary Γ_z in a positive way, then also $w = f(z)$ will follow the boundary in a positive way.

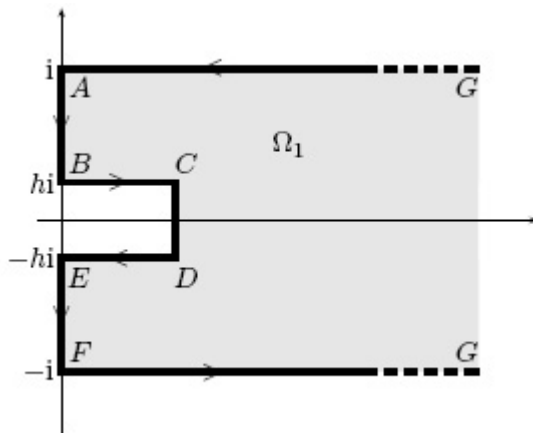


Figure 4.6: Right part of the foil identified with a subset of the complex plane.

Note that in theorem 4.5.1 the conformal map f is uniquely determined by the three conditions: $\operatorname{Re}(f(z_0)) = \operatorname{Re}(w_0)$, $\operatorname{Im}(f(z_0)) = \operatorname{Im}(w_0)$ and $\arg(f'(z_0)) = \alpha$.

We would like to use conformal mappings to map Ω_1 to the upper half plane, on which a solution is easier to compute. By the Schwarz-Christoffel formula [6] we can construct a conformal mapping f_1 from the upper half plane $\Omega_2 := \{w \in \mathbb{C} : \operatorname{Im} w > 0\}$ to the unbounded polygon Ω_1 (see figure 4.7),

$$f_1(w) = C_1 \int_0^w \frac{\sqrt{\tilde{w} - w_C} \sqrt{\tilde{w} - w_D}}{\sqrt{\tilde{w} - w_A} \sqrt{\tilde{w} - w_F} \sqrt{\tilde{w} - w_B} \sqrt{\tilde{w} - w_E}} d\tilde{w} + D_1,$$

where $C_1, D_1 \in \mathbb{C}$ are yet to be determined. The path of integration should be chosen in the upper half plane and f_1 is such that the points w_A, w_B, \dots, w_G are mapped onto the points A, B, \dots, G in Ω_1 . We have the freedom to choose three of the real points w_A, w_B, \dots, w_G . This will fix the remaining points. Let us take $w_G = \infty$, $w_B = -1$ and $w_E = 1$. Because of symmetry we have $w_D = -w_C$ and $w_F = -w_A$. Since $f_1(0) = L$, again by symmetry, we have

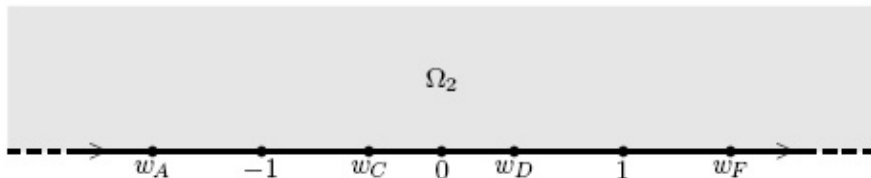


Figure 4.7: f_1^{-1} will map the unbounded polygon Ω_1 onto the complex upper half plane Ω_2 .

$D_1 = L$. This leads to

$$f_1(w) = C_1 \int_0^w \frac{\sqrt{\tilde{w} - w_C} \sqrt{\tilde{w} + w_C}}{\sqrt{\tilde{w} - w_A} \sqrt{\tilde{w} + w_A} \sqrt{\tilde{w} + 1} \sqrt{\tilde{w} - 1}} d\tilde{w} + L. \quad (4.31)$$

For C_1 and the negative numbers w_A and w_C we know that $w_A < -1$ and $-1 < w_C < 0$ and they can be determined from the following system of three equations with three unknowns:

$$f_1(w_A) = i, \quad (4.32a)$$

$$f_1(-1) = hi, \quad (4.32b)$$

$$f_1(w_C) = L + hi. \quad (4.32c)$$

The inverse function f_1^{-1} is also conformal and will map Ω_1 onto Ω_2 .

If there were only Dirichlet boundary conditions then a solution would be easy to find. The real part and the imaginary part of an analytic function $x + iy \mapsto \phi(x + iy)$ can be regarded as a harmonic function in x and y . The function $\arg(w - w_A)$ is the imaginary part of the analytic function $\ln(w - w_A)$ and will therefore be harmonic in the complex upper half plane. Furthermore it also satisfies the conditions $c = 1$ on the segment GA , and $c = 0$ on BE . However, the Neumann boundary condition $\partial c / \partial n = 0$ is not satisfied on AB and EF . Therefore we would like to map Ω_2 onto a domain where we can satisfy all boundary conditions.

Knowing w_A , w_C and C_1 , we can construct another Schwarz-Christoffel mapping $f_2 : \Omega_2 \rightarrow \Omega_3$ where Ω_3 has the geometry that is drawn in figure 4.8. The mapping will be of the form

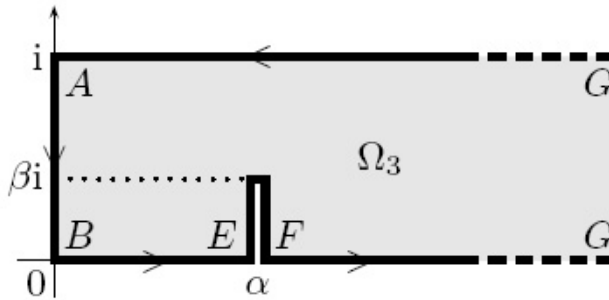


Figure 4.8: Geometry of Ω_3 .

$$f_2(w) = C_2 \int_0^w \frac{\tilde{w} - \gamma}{\sqrt{\tilde{w} - w_A} \sqrt{\tilde{w} + 1} \sqrt{\tilde{w} - 1} \sqrt{\tilde{w} + w_A}} d\tilde{w} + D_2, \quad (4.33)$$

where γ is some real number. Again the path of integration is in the upper half plane. This mapping maps Ω_2 to a set that has boundary angles $\frac{\pi}{2}$ at the points A , B , E and F . There exists a $\gamma \in \mathbb{R}$, such that E and F coincide and

the boundary of Ω_3 makes an angle of 2π at $\alpha + \beta i := f_2(\gamma)$, as shown in figure 4.8, and

$$\Omega_3 = \{z \in \mathbb{C} : \operatorname{Re}(z) > 0, 0 < \operatorname{Im}(z) < 1\} \setminus \{z \in \mathbb{C} : \operatorname{Re}(z) = \alpha, 0 < \operatorname{Im}(z) < \beta\}.$$

The lengths of the segments of the boundary, however, are not fixed yet. We determine C_2 , D_2 and γ from the three equations:

$$f_2(-1) = 0, \quad (4.34a)$$

$$f_2(w_A) = i, \quad (4.34b)$$

$$f_2(1) = f_2(-w_A). \quad (4.34c)$$

Now, B will be located in the origin in Ω_3 , A will be at i and the points E and F coincide in Ω_3 , like in figure 4.8. Note that automatically $f_2(\infty) = \infty$.

The real part and the imaginary part of an analytic function $x + iy \mapsto \phi(x + iy)$ can be regarded as a harmonic function in x and y . The imaginary part \tilde{c} of the analytic function $\phi(z) = z$ is equal to $\operatorname{Im}(z) = y$ and satisfies the conditions $\tilde{c} = 1$ on the segment GA , $\partial\tilde{c}/\partial n = 0$ on AB , $\tilde{c} = 0$ on BE , $\partial\tilde{c}/\partial n = 0$ on EF and $\tilde{c} = 0$ on FG . Therefore $\tilde{c}(x, y) = y$ solves the boundary value problem on $I(\Omega_3)$, where $I : \mathbb{C} \rightarrow \mathbb{R}^2$ is given by

$$I(x + iy) = (x, y).$$

We have a conformal mapping $f : \Omega_1 \rightarrow \Omega_3$, namely

$$f = f_2 \circ f_1^{-1}.$$

Compositions of harmonic functions with conformal mappings are again harmonic functions. The solution $c : \Omega \rightarrow \mathbb{R}$ of the original problem is therefore given by

$$c = \tilde{c} \circ I \circ f \circ I^{-1} = \operatorname{Im} \circ f \circ I^{-1}.$$

In other words

$$c(x, y) = \operatorname{Im} (f_2(f_1^{-1}(x + iy))). \quad (4.35)$$

Results

Using conformal mappings we have reduced solving boundary value problem (4.30) to solving the two systems of equations (4.32) and (4.34). These can be solved numerically and possibly even analytically. Once the unknown constants are found, equation (4.35) will yield the oxygen concentration and thus the permeating oxygen can be calculated.

To test which scavenger configuration is better, we need some kind of measure for the oxygen exposure. Of course a configuration is better if the total flux of oxygen through Γ_3 (see figure 4.5) becomes smaller. Thus we would like to minimize:

$$\int_{-\infty}^{\infty} \frac{\partial c}{\partial \mathbf{n}}(x, -1) \, dx = 2 \int_0^{\infty} \frac{\partial c}{\partial \mathbf{n}}(x, -1) \, dx.$$

However, this integral could be divergent. For large x the oxygen flux will not be affected by the presence of the block, therefore a suitable alternative could be the following: for $s > 0$ introduce the average flux F_s defined as

$$F_s(L, h) := \frac{1}{s} \int_0^s \frac{\partial c}{\partial \mathbf{n}}(x, -1; L, h) \, dx.$$

Now for a given area C and fixed s large enough we can compare different configurations with $L * h = C$ by computing for which configuration F_s is smallest.

The previous analysis can also be performed for a bounded strip rather than an infinite strip. The advantage of such a model is that periodic boundary conditions can be assumed on both ends and thus a whole series of scavenger blocks can be modelled. However, the Schwarz-Christoffel formula f_1 in (4.31) would get an additional constant that has to be determined. This would complicate the analysis slightly because now we get an extra Neumann boundary condition and four equations with four unknowns have to be solved rather than three equations with three unknowns. This approach is more realistic since the foil contains more than one scavenger droplet, whereas only minor extra complications arise.

Finally, we could also consider the two limiting cases $(L, h) \rightarrow (0, C)$ and $(L, h) \rightarrow (C, 0)$. These two cases show the extremes of stretching in the horizontal and vertical direction and could already give an indication which kind of stretching is better. Moreover, it simplifies the analysis. In these two cases only two equations have to be solved to find Ω_2 . Also this simplification might allow us to solve the integral expression for f_1 explicitly, so that the system (4.32) transforms into a system of two algebraic equations. Calculating f_2 remains as difficult as it was.

4.6 One-dimensional numerical simulation

We have done simulations in one and two space dimensions. The one-dimensional experiments are described in this section, the two-dimensional simulations in section 4.7. Full numerical simulations of the three-dimensional model turned out not be feasible in the one week period of the Study Group. Even with more time available it is not certain that a three-dimensional model can be computed with accuracy within a reasonable time period.

In order to find numerical approximations we have to make some assumptions and set some boundary and initial conditions. For both the one- and two-dimensional simulations we have taken the following conditions.

- The time domain has been scaled to $[0, 1]$. The space domain has been scaled to $\Omega = [0, 1]$ for the one-dimensional simulations and $\Omega = [0, 1] \times [0, 1]$ for the two-dimensional simulations (see section 4.7).
- $c(x, 0) = 0$, at the start there is no oxygen in the material

- $c(0, t) = 0$, $c(1, t) = c_a$. At the food boundary of the foil, i.e. $x = 0$, the oxygen concentration is zero; at the other boundary the concentration of oxygen is constant.
- $s(x, 0) = \phi(x)$, the initial concentration of scavenger material is a prescribed function of the position.

The initial scavenger concentration should describe the scavenger material that is present in the droplets in the foil. These droplets are spherical when the foil is created. However the foil can also be stretched in the fabrication process. This stretching can occur in either one or two directions leading to cigar or pancake shaped droplets, respectively. In the numerical simulations we have used rectangular bump functions for the initial scavenger concentrations. This choice was done for convenience but other initial concentrations (such as perfect spheres, cigars or pancakes) can also be analyzed numerically. An argument for our choice of rectangular bump functions is that the partial differential equation used to model the process is a diffusion equation with a reaction term. The diffusion term has the property that all solutions will be smooth (even for non-smooth initial conditions).

Topological effects

We expect that the two- and three-dimensional models will be quite different from the one-dimensional model for topological reasons. In figure 4.9 we have a schematic picture of the foil for the two-dimensional model. The scavenger droplets are indicated as black spots. The oxygen particles can go through the foil in various paths. We have drawn three different type of paths in the figure. The dotted path represents an oxygen particle that enters the foil, but is absorbed by a scavenger particle. The dashed path is the path of a particle that enters the foil, passes through a droplet but is not absorbed. Finally the oxygen particle reaches the food boundary. In solid black there are two paths where the oxygen particle passed through the foil without encountering any scavenger material. The black paths cannot occur in the one-dimensional model. There every oxygen particle that passes through the foil will have to pass through one or more droplets (assuming that there is at least one droplet).

One-dimensional simulations

The one-dimensional simulations are important to get a feeling for the possible three-dimensional results. For various initial configurations of scavenger material a numerical solution was calculated using Mathematica. The function `NDSolve` has been used to find a numerical approximation to the partial differential equation.

The initial scavenger configurations that have been analyzed are

- A homogeneous scavenger concentration.
- Bump functions with a various number of bumps.

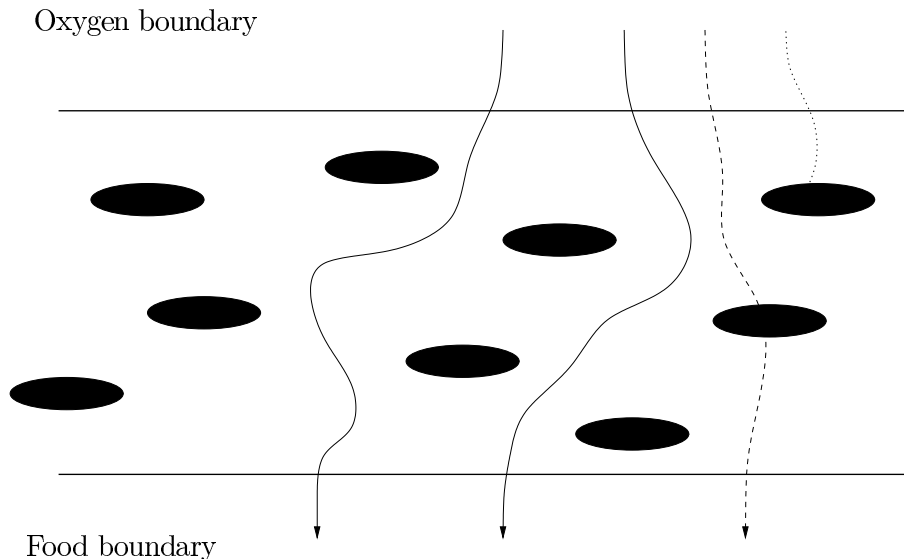


Figure 4.9: Oxygen paths in the foil

During the simulations we found out that the results depend very sensitively on the values of the parameters. We will give a few examples that illustrate the various phenomena that can occur. In the simulations described below we have taken $\alpha = \beta = 1$, $D = 1$, $\kappa_c = 50$, $\kappa_s = 1$ and $c_a = 1$.

Four bump scavenger configuration

In figure 4.10 we have plotted the initial scavenger concentration for a four-bump scavenger concentration. Given the initial conditions described above we can find a numerical solution to the system of partial differential equations (4.2a)-(4.2b) for this particular initial scavenger concentration. In figure 4.11 we have plotted the scavenger configuration as a function of time and position. We can see that one by one the droplets of scavenger material are shrinking because of absorption of oxygen.

High reaction rate limit

In the limit of a very high reaction rate the scavenger material reacts almost instantly with the oxygen. In this situation no oxygen can penetrate the foil until all the scavenger material has been absorbed. The absorption rate of the scavenger material is determined by the distance of the scavenger material to the oxygen boundary. Here we can say the the best distribution of scavenger material is placing all the scavenger material close to the food boundary.

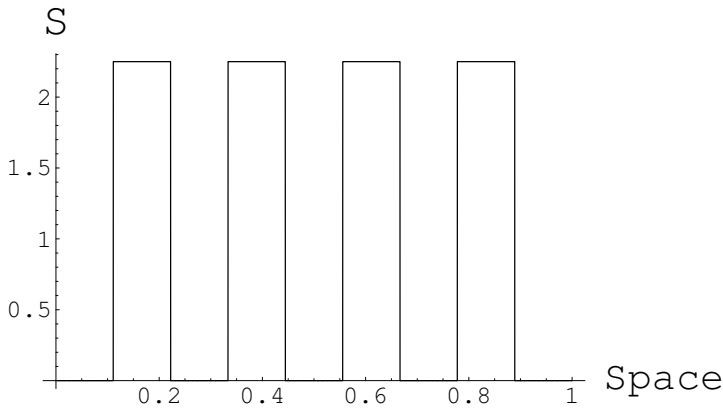


Figure 4.10: Initial scavenger concentrations in the four-bump configuration

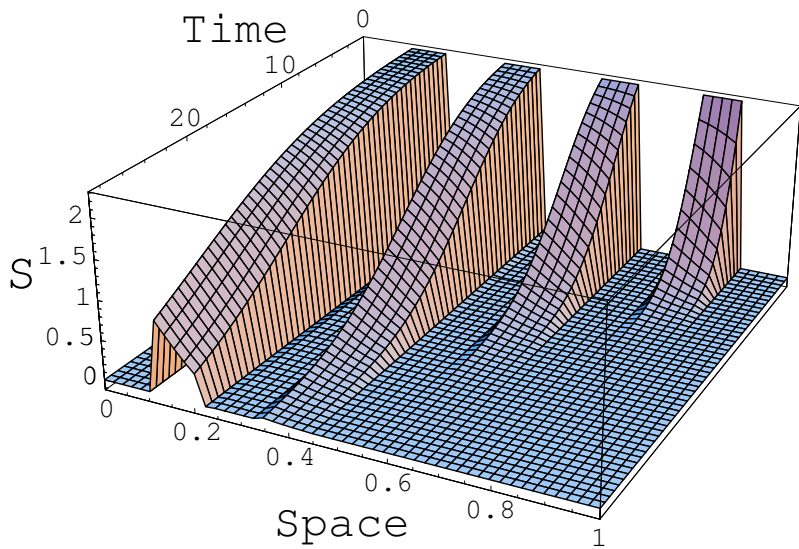


Figure 4.11: Scavenger concentrations for the four-bump configuration

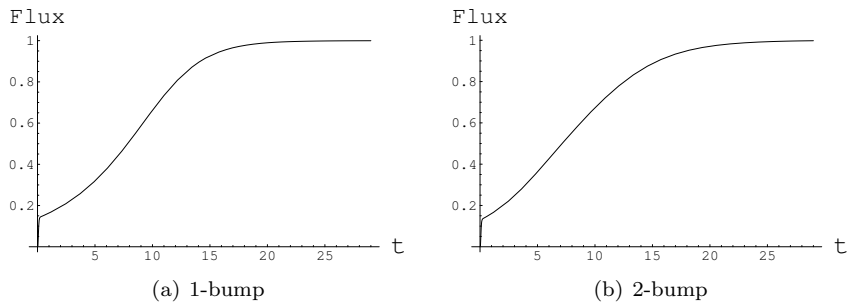


Figure 4.12: Flux for various scavenger configuration

Time scales

We compare a one bump scavenger concentration with a two bump scavenger concentration. The flux of oxygen has been plotted for both configurations and also the difference in flux. This flux (and the integral of the flux over time) is a measure for the amount of oxygen that reaches the food and hence a measure of how good the foil is.

In figure 4.12 the flux for the one-bump configuration was plotted. In the plot for the flux we can see 3 time domains:

Oxygen spreading through the foil This happens very quickly and is a consequence of the diffusion equation used to model the process. In the plot above the timescale is roughly 0.1 time units.

Reaction The scavenger material reacts with the oxygen that diffuses through the foil. During this period the amount of scavenger material decreases and as a result the flux and concentration of oxygen in the foil increases.

Final state For this simulation the final state is reached after roughly 20 time units. All the scavenger material has reacted with the oxygen and the numerical solution approaches the exact solution in the case of no scavenger material.

The flux for the one-bump and two-bump configurations looks very similar. However if we plot the difference of flux we get an interesting picture. In figure 4.13 the difference is shown. We can see that at the start, up to $t \approx 1.5$, the flux for the one-bump configuration is higher. Hence the foil with a two-bump configuration performs better than the foil with the one-bump configuration. From $t \approx 1.5$ to $t \approx 10.6$ the one-bump foil performs better. For larger values of t the two-bump performs better. In the limit the scavenger material in both foils has reacted away and the flux in both foils becomes identical. If we integrate the flux we get the total amount of oxygen that has passed through the foil. The integral of the flux difference from figure 4.13 is plotted in figure 4.14. Here we see that also for the total amount of oxygen the

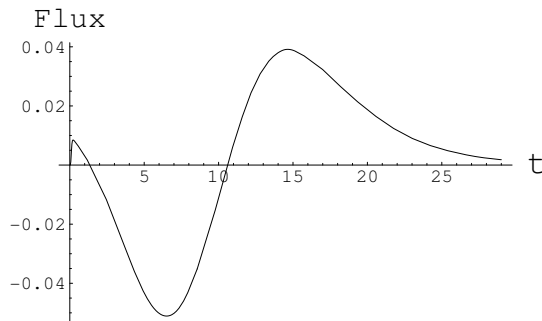


Figure 4.13: Flux difference between one-bump and two-bump scavenger configuration

one-bump configuration performs better for small time, but worse for t between 2.5 and 20.5 and again better for t larger than 20.5.

The results may seem surprising but can give an intuitive explanation as follows. Scavenger material is more effective if it is closer to the oxygen boundary. From the one-bump and two-bump configurations, the bump close to the oxygen boundary in the two-bump configuration is closest to the oxygen boundary. Therefore for small t this bump will make the two-bump configuration perform better. However, since this bump is close to the oxygen boundary it also reacts faster with the oxygen than the other bumps. After some time this bump will have been reduced by reaction and the bump in the one-bump configuration becomes dominant; this makes the one-bump configuration perform better. If also this bump is reacted away for the most part the last bump, the bump close to the food boundary in the two-bump configuration, dominates and makes the two-bump foil again perform better. The explanation above is only an intuitive one: for other configurations only a numerical calculation can give the flux and total amount of oxygen as a function of time.

4.7 Two-dimensional numerical simulation

two-dimensional numerical simulations

Simulations in two-dimensions were made in order to investigate the difference between shapes of scavenger droplets and oxygen flux through the film. We model the film as an infinitely long strip consisting of one layer of rectangular cells (see Figure 4.15). Each cell has a rectangular scavenger particle in the center and the particle occupies 10% of the cell. At the outer side of the film there is normal air and the oxygen concentration is c_a ; at the inner side the oxygen concentration is 0 (oxygen immediately reacts with the food). Initially there is no oxygen in the film.

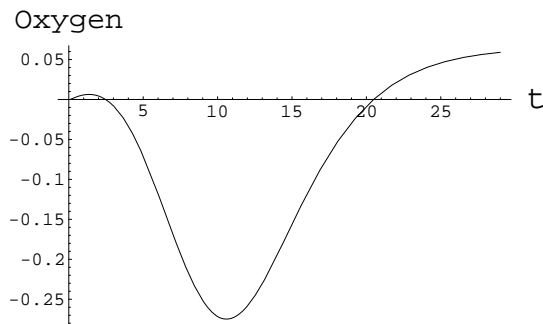


Figure 4.14: Difference between the total amount of oxygen passed between one-bump and two-bump scavenger configuration

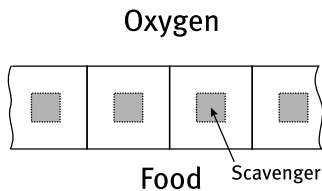


Figure 4.15: Two-dimensional model of the film with scavenger droplets. The film consists of one layer of rectangular elements. Each element has one scavenger particle in the center.

We proceed with a cell of length A and thickness B with the rectangular scavenger particle of the length a and the thickness b (see Figure 4.16). The cell

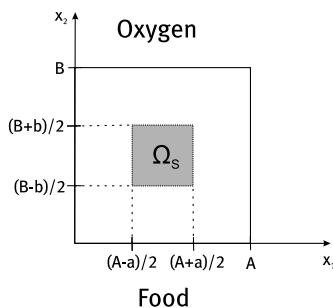


Figure 4.16: Cell with the scavenger particle in the center.

occupies the region $\Omega = \{\mathbf{x} \in \mathbb{R}^2 : 0 \leq x_1 \leq A, 0 \leq x_2 \leq B\}$ and the scavenger particle occupies $\Omega_s = \{\mathbf{x} \in \mathbb{R}^2 : (A-a)/2 \leq x_1 \leq (A+a)/2, (B-b)/2 \leq x_2 \leq (B+b)/2\}$. In the equations (4.2a)-(4.2b) we assume that $\alpha = \beta = 1$.

Therefore, the reaction-diffusion process in the cell is described by

$$\frac{\partial c(\mathbf{x}, t)}{\partial t} = D\Delta c(\mathbf{x}, t) - \kappa s(\mathbf{x}, t)c(\mathbf{x}, t), \quad (4.36a)$$

$$\frac{\partial s(\mathbf{x}, t)}{\partial t} = -\kappa s(\mathbf{x}, t)c(\mathbf{x}, t), \quad (4.36b)$$

where $\mathbf{x} \in \Omega$ and $t \in [0, \infty)$. The initial concentration of oxygen in Ω is zero

$$c(\mathbf{x}, 0) = 0, \quad \mathbf{x} \in \Omega, \quad (4.37)$$

and the initial concentration of scavenger is s_0 in Ω_s and zero outside Ω_s .

$$s(\mathbf{x}, 0) = \begin{cases} s_0, & \mathbf{x} \in \Omega_s, \\ 0, & \mathbf{x} \in \Omega \setminus \Omega_s. \end{cases} \quad (4.38)$$

Because the concentrations of oxygen at both sides of the film are constant, we have Dirichlet boundary conditions for c at the top and bottom of the cell

$$c(\mathbf{x}, t)|_{x_2=0} = 0, \quad c(\mathbf{x}, t)|_{x_2=B} = c_a. \quad (4.39)$$

At the lateral sides of the cell we impose the periodic boundary condition

$$c(\mathbf{x}, t)|_{x_1=0} = c(\mathbf{x}, t)|_{x_1=A}, \quad \left. \frac{\partial c(\mathbf{x}, t)}{\partial x_1} \right|_{x_1=0} = \left. \frac{\partial c(\mathbf{x}, t)}{\partial x_1} \right|_{x_1=A}, \quad (4.40)$$

because the film consists of infinitely many cells.

We scale the distance x to the thickness, B , of the film, $\tilde{\mathbf{x}} = \mathbf{x}/B$, the time $\tilde{t} = tD/B^2$, the oxygen concentration as $\tilde{c} = c/c_a$, the scavenger concentration $\tilde{s} = s/c_a$ (here we assume that s and c have the same dimensions). After scaling the system (4.36a)-(4.40) becomes

$$\frac{\partial \tilde{c}(\tilde{\mathbf{x}}, \tilde{t})}{\partial \tilde{t}} = \Delta \tilde{c}(\tilde{\mathbf{x}}, \tilde{t}) - \tilde{\kappa} \tilde{s}(\tilde{\mathbf{x}}, \tilde{t}) \tilde{c}(\tilde{\mathbf{x}}, \tilde{t}), \quad (4.41a)$$

$$\frac{\partial \tilde{s}(\tilde{\mathbf{x}}, \tilde{t})}{\partial \tilde{t}} = -\tilde{\kappa} \tilde{s}(\tilde{\mathbf{x}}, \tilde{t}) \tilde{c}(\tilde{\mathbf{x}}, \tilde{t}), \quad (4.41b)$$

$$\tilde{c}(\tilde{\mathbf{x}}, 0) = 0, \quad \tilde{\mathbf{x}} \in \tilde{\Omega}, \quad (4.41c)$$

$$\tilde{s}(\tilde{\mathbf{x}}, 0) = \begin{cases} \tilde{s}_0, & \tilde{\mathbf{x}} \in \tilde{\Omega}_s, \\ 0, & \tilde{\mathbf{x}} \in \tilde{\Omega} \setminus \tilde{\Omega}_s, \end{cases} \quad (4.41d)$$

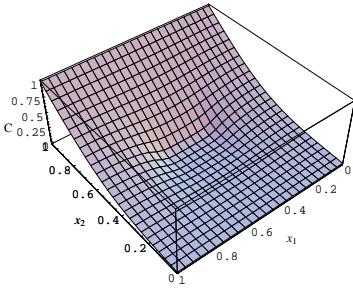
$$\tilde{c}(\tilde{\mathbf{x}}, \tilde{t})|_{\tilde{x}_2=0} = 0 \quad (4.41e)$$

$$\tilde{c}(\tilde{\mathbf{x}}, \tilde{t})|_{\tilde{x}_2=1} = 1 \quad (4.41f)$$

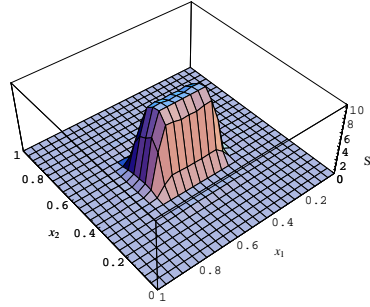
$$\tilde{c}(\tilde{\mathbf{x}}, \tilde{t})|_{\tilde{x}_1=0} = \tilde{c}(\tilde{\mathbf{x}}, \tilde{t})|_{\tilde{x}_1=\tilde{A}}, \quad (4.41g)$$

$$\left. \frac{\partial \tilde{c}(\tilde{\mathbf{x}}, \tilde{t})}{\partial \tilde{x}_1} \right|_{\tilde{x}_1=0} = \left. \frac{\partial \tilde{c}(\tilde{\mathbf{x}}, \tilde{t})}{\partial \tilde{x}_1} \right|_{\tilde{x}_1=\tilde{A}}, \quad (4.41h)$$

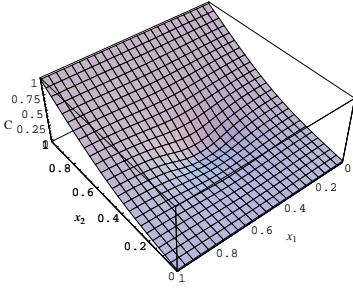
for $\tilde{\mathbf{x}} \in \tilde{\Omega}$ and $\tilde{t} \in [0, \infty)$. Here $\tilde{A} = A/B$, $\tilde{a} = a/B$, $\tilde{b} = b/B$, $\tilde{\Omega} = \{\mathbf{x} \in \mathbb{R}^2 : 0 \leq x_1 \leq \tilde{A}, 0 \leq x_2 \leq 1\}$, $\tilde{\Omega}_s = \{\mathbf{x} \in \mathbb{R}^2 : (\tilde{A} - \tilde{a})/2 \leq x_1 \leq (\tilde{A} + \tilde{a})/2, (1 - \tilde{b})/2 \leq$



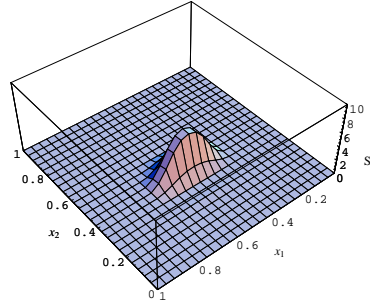
(a) Oxygen concentration at $t = 0.1$.



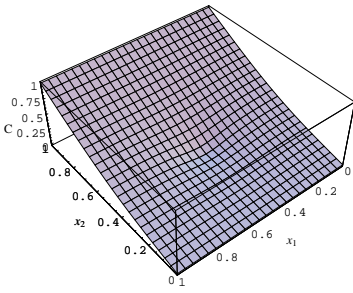
(b) Scavenger concentration at $t = 0.1$.



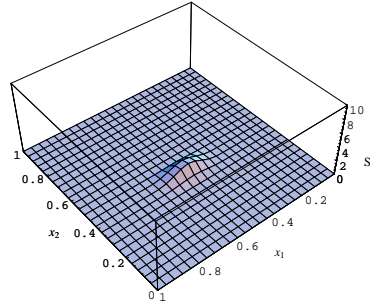
(c) Oxygen concentration at $t = 0.5$.



(d) Scavenger concentration at $t = 0.5$.



(e) Oxygen concentration at $t = 0.7$.



(f) Scavenger concentration at $t = 0.7$.

Figure 4.17: Concentrations of oxygen c and scavenger s in the cell at times $t = 0.1$, $t = 0.5$ and $t = 0.7$ for $a = 0.4$ and $b = 0.25$.

$x_2 \leq (1 + \tilde{b})/2\}$, $\tilde{\kappa} = \kappa c_a B^2/D$ and $\tilde{s}_0 = s_0/c_a$. In the following we omit the tildes.

The system (4.41a)-(4.41h) is solved numerically in Mathematica for $\kappa = 100$, $s_0 = 10$ and $A = 1$. We vary the shape of the scavenger particle while the area remains constant, $ab = 0.1$, because the scavenger occupies 10% of the film. In Figure 4.17 we present the oxygen concentrations and the scavenger concentrations in the cell at times $t = 0.1$, $t = 0.5$ and $t = 0.7$. The size of the scavenger particle is taken to be $a = 0.4$, $b = 0.25$. When the scavenger disappears the oxygen profile becomes more straight.

For a real application it is important to know a flux through the film. We compute the flux through the cell as

$$F(t) = - \int_0^A \frac{\partial c(x_1, 0, t)}{\partial x_2} dx_1. \quad (4.42)$$

In Figure 4.18 we present the flux for different sizes of the scavenger particle (see Figure 4.19) as a function of time. If the length of the scavenger particle

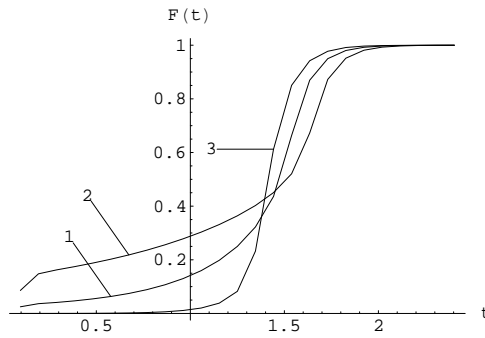


Figure 4.18: Flux through the cell for different sizes of the scavenger particle (line 1 $a = 0.6666$, $b = 0.15$; line 2 $a = 0.3162$, $b = 0.3162$; line 3 $a = 1$, $b = 0.1$).

is equal to the length of the cell (line 3) then the short time behavior is better than that for the droplets with other shapes. But after the scavenger disappears the flux becomes large, while the droplets with the other shapes (lines 1 and 2) still react with the oxygen. If the scavenger particle has a square shape (line 2) then it reacts longer than the droplets with the other rectangular shapes. The flux of a square scavenger droplet is initially larger than that of scavenger droplets with other rectangular shapes.

4.8 Conclusions

This Study Group turned out to be an inspiring and creative week. Not only is this our perception as a group, but it also comes forward in the scientific

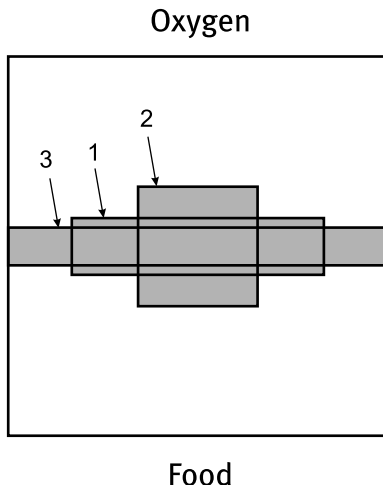


Figure 4.19: Different shapes of the scavenger particle (line 1 $a = 0.6666$, $b = 0.15$; line 2 $a = 0.3162$, $b = 0.3162$; line 3 $a = 1$, $b = 0.1$).

contents (and size!) of this report and the various approaches that are proposed within. The fruitful cooperation is moreover underlined by the large number of authors who were willing to contribute, and the number of people that helped us during the week. Different approaches were discussed critically, with the result that only a small part of the ideas coined were found to be worth reporting. These approaches cover methods, techniques and ideas about three general areas: the modelling (section 4.2), the analysis (section 4.3, 4.4 and 4.5), and numerical simulations (section 4.6 and 4.7). We give a short summary of the conclusions that we have drawn.

The stochastic model that is derived in section 4.2 converges in a weak sense to the model proposed by DSM. In section 4.3 it is concluded that for very small length scales and for small amounts of scavenger, thin, elongated droplets are more effective than small spherical droplets. In section 4.4 it is shown that, in the one-dimensional case, the penetration time of oxygen through the foil depends quadratically on the diameter of scavenger particles, and linearly on the amount of scavenger, and linearly on the distance between the particles and the outside surface of the foil. In section 4.5 it is concluded that in order to investigate the influence of droplet shapes, the DSM model can be converted via conformal mappings to a set of equations that are more attractive numerically and analytically. One-dimensional simulations in section 4.6 show that two small homogeneously distributed droplets are more effective than one big droplet. Finally, two-dimensional simulations in section 4.7 show that cigar- and pancake-shaped droplets perform better than spherical droplets initially, but that then the scavenger reacts away faster.

The problem owners from DSM, prof. dr. Han Slot, and dr. Alexander Stroeks were very clear in their explanation and motivation of the problem, which helped us work in the most interesting directions, mathematically as well as practically. We do hope that the mathematical insight we developed and reported, will provide guidelines in ways of solving their problems.

4.9 Acknowledgement

We acknowledge the input and time of Alexander Stroeks and Han Slot from DSM and the work of all the participants in the DSM group and the people from outside our group who helped by supplying new ideas: Hala Elrofai, Vincent Guyonne, Joost Hulshof, Vivi Rottschäfer, Martin van der Schans, Paul Zegeling, Geertje Hek, Remco van der Hofstad, and Matthias Röger.

4.10 Bibliography

- [1] D. Hilhorst, R. van der Hout, and L. A. Peletier. The fast reaction limit for a reaction-diffusion system. *J. Math. Anal. Appl.*, 199(2):349–373, 1996.
- [2] U. Hornung. *Homogenization and Porous Media*. Springer-Verlag, Berlin, 1996.
- [3] L.P.B.M. Janssen J.M.Smit, E. Stammers. *Fysische Transportverschijnselen I*, volume III. Delft University Press, 6 edition, 1973.
- [4] Landim C. Kipnis C. *Scaling limits of interacting particle systems*. Springer, Berlin, 1999.
- [5] T.M. Liggett. *Interacting particle systems*. Springer, New York, 1985.
- [6] Z. Nehari. *Conformal Mapping*. McGraw-Hill, New York, 1952.

RADIOACTIVE NEEDLEWORK

Reconstruction of needle-positions in radiation treatment

Claude Archer¹, Frits van Beekum², Andrew Hill³,
Michiel E. Hochstenbach⁴, and Ionica Smeets⁵

Abstract

Nucletron presented a medical problem to the SWI 2006: how to find needles used for cancer treatment in a prostate? More concretely: how to find the positions of these needles from distorted images from an ultrasound probe? Section 1 explains the background of this problem. In Section 2 we deal with physical explanations for the distortions. In Section 3 we give a brief overview of medical imaging and explain which techniques we used to clean up the images.

5.1 Introduction

Before we state the problem posed by Nucletron, we explain the background of their problem.

Brachytherapy

In prostate cancer treatment a new therapy has come in use, where, unlike the external radiation in the well-known chemotherapy, the radiation is now supplied by a number of tiny sources *inside* the prostate. Nucletron, with headquarters in Veenendaal, the Netherlands, has developed this *brachytherapy*—brachys is the Greek word for short (here: with respect to distance)—and has been using this medical technology for over five years now.

The short distance between source and target enables a considerable reduction of the dose, thus reducing the risk of damaging the surrounding tissue. Moreover, the distribution of the sources over the prostate is optimised so as to create a prescribed level of radiation over the prostate interior, a higher level

1: Haute École Francisco Ferrer, 2: Universiteit Twente, 3: University of Bath, 4: Case Western Reserve University, 5: Universiteit Leiden

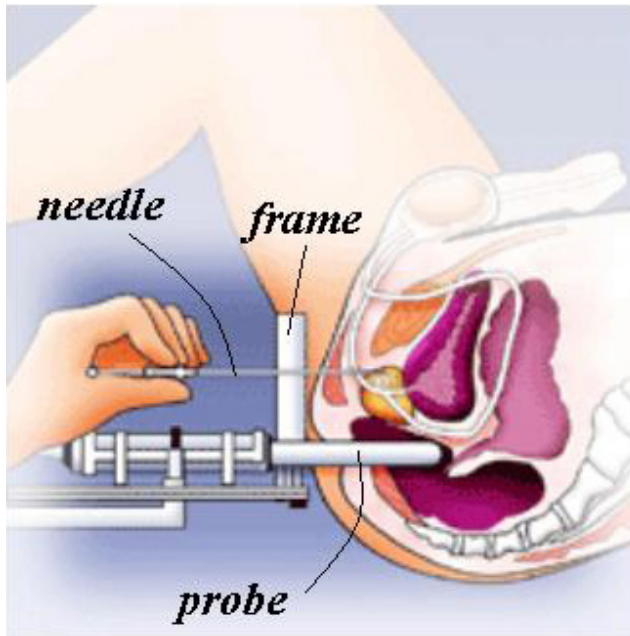


Figure 5.1: Position of the device with frame for inserting the needles and probe for ultrasound monitoring.

near the outer surface, and a lower level near the urethra (which runs right through the prostate).

The sources, so-called seeds, look like pieces of thin pencil-lead and are brought into position through hollow needles with outer diameter 1.25 mm. Typically 2 to 6 seeds are placed in one needle, separated by a dummy or a thread of biodegradable material. The number and spacing of the seeds in each needle and the optimal position for each needle are computed before the operation starts. When the needle has been put in the prostate, this train of seeds is pushed out, while at the same time the needle is withdrawn, thus leaving the seeds and spacers in the desired positions.

The radioactive decay of the used iodine or palladium seeds is such that in half a year the radiation is no more than a few percent. And, although this treatment is not the answer to all types of prostate problems, patients who have been selected for brachytherapy may have good hope that the tumors will disappear and may stay away for 15 years and more.

The problems during the operation

The prostate has the shape and size of a firm walnut, and will be perforated by 12 to 30 needles. The access is through the perineum, which is the skin and tissue between scrotum and anus. A frame containing the needles is positioned

to this perineum. What makes the operation difficult is that the needles, when being pushed in, will not follow the “mathematical” path which is a straight line exactly perpendicular to the frame. In practice, a needle may deviate and make a curve, e.g., because it meets density differences in the tissue or because of play in the frame. Moreover, pushing a needle will affect the position and shape of the prostate itself, so the second needle cannot take the intended position in the tissue even when it would be perfectly perpendicular to the frame.

Due to all this, the calculated optimal distribution of seeds over the needles has become less informative once the needles have been pushed in. Therefore, each time before inserting a needle, the new geometry of the prostate and the position of the needles are measured first. As a result it may be decided to insert one or more extra needles in poorly covered areas. After all this the optimisation procedure is run again and the seed trains are pushed in. To know where the needles are in the body an ultra-sound sensor is used during the treatment.

Running the optimising software takes only one or two minutes, but measuring the new situation is the bottleneck, as it may take approximately fifteen minutes. This is thought to be too long, not only for the doctors and staff who will stand idle in waiting, but also for the patient: the period under anaesthesia should be kept as short as possible. Moreover, due to the perforation the tissue will start swelling, and in fifteen minutes the measured data might have become unreliable. This is another motivation to minimize the measuring time.

Ultrasound

Measuring the initial situation in the body and monitoring the actions during the operation is done by *ultrasonography* or *ultrasound*. This is a medical imaging technique that uses high frequency sound waves and their echoes. The technique is similar to the echolocation used by bats, whales and dolphins, as well as SONAR used by submarines.

In short, it works as follows*. From a transmitter at a given point, high-frequency (here: 5 to 7.5 megahertz) sound pulses are sent out into the body. The sound waves travel out and hit a boundary between tissues (e.g., between fluid and soft tissue, soft tissue and needle). Here some of the sound waves get reflected back to the transmitter (which itself can act as a receiver or sensor), while some travel on further until they reach another boundary and get reflected. The reflected waves are picked up by the sensor and relayed to the machine. The machine calculates the distance from the transmitter/sensor to the tissue or organ (boundaries) using the speed of sound in tissue, 1540 m/s, and the time of the echo’s return (usually on the order of millionths of a second). The machine displays the distances and intensities of the echoes on the screen, forming a two dimensional image like the ones shown in Figure 5.1.

*See <http://electronics.howstuffworks.com/ultrasound1.htm>

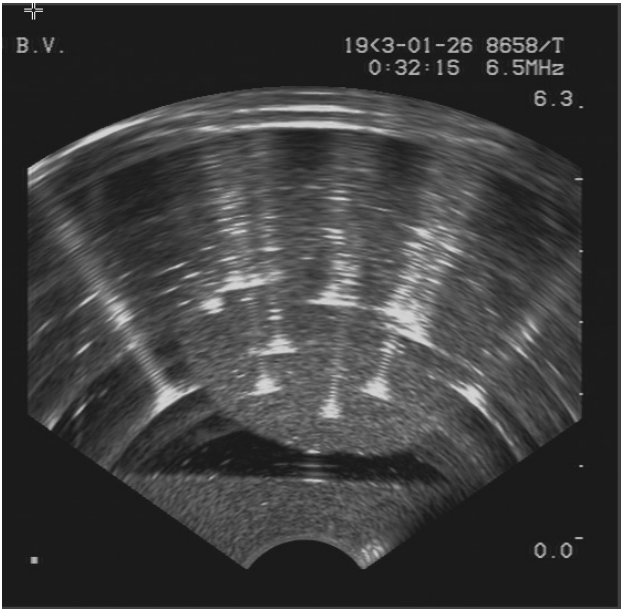


Figure 5.2: A transversal image of the prostate with 12 needles in it. Even the experts at Nucletron had difficulties in identifying them all.

In scanning the prostate, a cylinder-shaped probe of 2 cm diameter is inserted in the rectum. An array of 96 miniature transmitter/sensors covers one quarter of a ring on this cylinder. As each transmitter is assumed to send out signals in a very confined direction perpendicular to the cylinder axis, the probe will scan a 90 degree sector in a plane (slice) transversal to the cylinder. Most of the pictures shown here are transversal scans. A sequence of transversal scans collected by moving the probe up and down in the rectum, can provide 3-dimensional information, from which also other than transversal images can be derived.

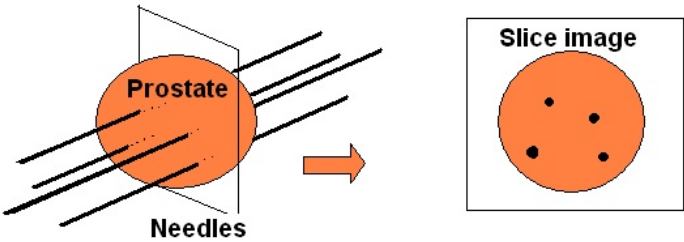


Figure 5.3: Obtaining transversal views of the prostate.

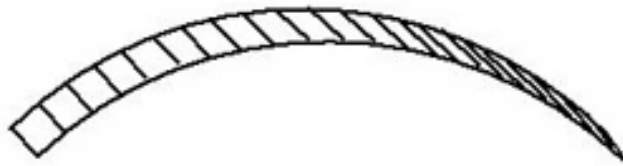


Figure 5.4: An array of 96 transmitters/sensors making one quarter of a “ring” around the cylindrical probe.

The problem as posed by Nucletron

A typical one-slice scan after inserting a dozen of needles is shown in Figure 5.1. What we want to identify is

- the boundary of the prostate, and
- the location of the needles.

The boundary of the prostate is rather easily seen as a more or less circular interface between two grey levels. Identifying the needles is more difficult. We see several bright spots that no doubt represent a needle, but from further inspecting the figure we can make the following three remarks:

- apart from individual needles we also see some larger bright areas that seem to include two or even three needles (a *cluster*);
- in the “shade” of a needle we see a series of ripples, while there is no physical object there (we will call them *artifacts*);
- if a needle is in the shade of another needle, we cannot see it. This is only natural, but in this case the artifacts seem to make it even more invisible.

With respect to the last remark there is no other remedy than trying to “look around” the nearby needle by changing the position of the probe.

The other two remarks are the core of the questions that Nucletron posed:

- are the artifacts a result of unwanted, but physically real, waves? If so, then what could we physically do about them, rather than suppressing them afterwards by image processing?
- can image processing help to clean up the picture, for instance by suppressing the background noise, decomposing clusters into single needles, suppressing the artifacts, etc., with the ultimate goal to end up with clear needle locations only?

In the next section we go into the details of possible physical explanations for the artifacts. In the last section we deal with medical imaging.

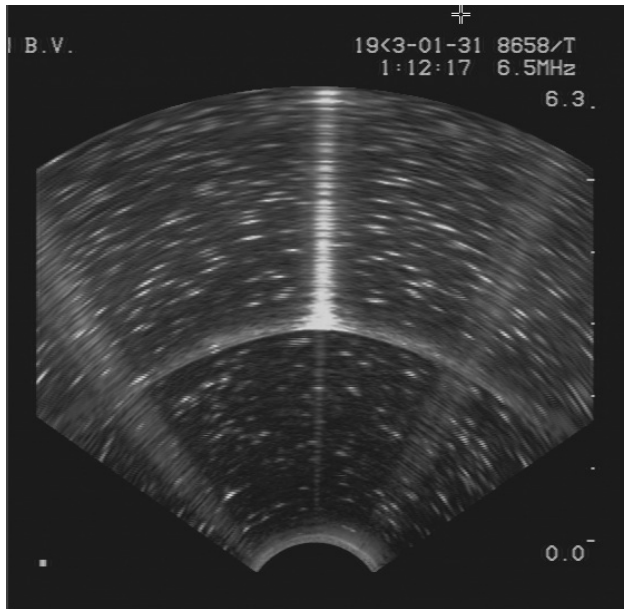


Figure 5.5: Even in a one needle situations artifacts occur.

5.2 Physical explanations for the artifacts

For interpretation of the images we have to think of the way the ultrasound works. Each transmitter sends a pulse, and from the time that passes until it receives the echo, it calculates the distance to the reflecting object. A sensor receiving a signal can only draw one conclusion: in the specific direction of this transmitter/sensor there is an object, and its distance is given by the time delay of the signal and the velocity of sound in the medium. Even if the signal would have had multiple reflections on several needles, the interpretation is still a mirroring object on this specific radial line.

However, the multiple needle reflections that we mentioned can not be the cause, as we asked Nucletron to scan a *one needle* situation in the form of a model in a laboratory set-up (i.e., a so-called *phantom* in water). The result is in Figure 5.2, where we still see the same type of artifacts.

Secondly, now restricting to the one needle case, we could think the artifacts might be an interference pattern from two (or more) neighbouring transmitters, reflecting at the same needle. However, the reality of an ultrasound image is that it is a composition of 96 separate images, each showing the echo of *one single transmitter* being active, while all others are silent *and* deaf. So the sensor only receives echoes of its own emitted pulse, no others.

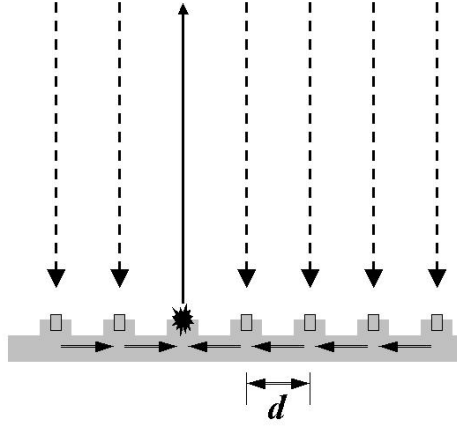


Figure 5.6: Echoes may cause vibrations propagating along the base.

Two arguments left

Knowing that only one transmitter/sensor is active, and knowing that the spatial pattern of artifacts must be interpreted as signals coming in at subsequent times, the question now is: is there, apart from direct reflection at the needle, any path that a signal can go such that it arrives back at the one and only active sensor at a later time? Can a pulse signal create some vibrations in the region of interest, which will start sending out periodic signals? We guess that here not the soft tissue, but the solid material can come into play: a needle and/or the ring-shaped structure where the sensors are assembled. We start with the latter.

Vibrations of the probe

If a pulse hits the needle, it will be reflected not only in the direction it originated from, but in a beam of directions. So the reflected signal will not only hit the transmitter it came from, but also the neighbouring ones. Now the neighbouring ones are deaf, so they do not receive the reflected signal. Nonetheless, the reflected signal may make vibrations in the socle (an elevation on which the sensor is mounted). And through the base of the mechanical structure the vibrations will arrive at the active sensor.

A first indication whether this phenomenon can be responsible for the artifacts is in a comparison of the periodicities. The distance d between two neighbouring sensors is $1/96$ of one quarter of the perimeter of the probe, which has 2 cm diameter. So $d = (\pi/4)/96$ times 20 mm, is about 0.16 mm. Now the velocity of sound in the base structure is three times higher than in the tissue. So in the scan image the stripings must have a distance of one third

of 0.16 mm, that is about 0.05 mm. About 600 of these fit in one prostate diameter of (say) 3 cm. However, just by counting in a scan image we find some 50 stripings go in one prostate diameter. So we must conclude that vibrations in the probe do not explain the artifacts.

Remark. Maybe the reflected signal could be prevented from hitting neighbouring transmitters if they themselves would send out an exactly counter-phased signal. However, phase control is hardly found in medical applications yet. In cardiology, working with a phased array is in development, but not yet operational.

The ratio 600/50, which is 12, leads us to the idea to search for vibrations on a length scale which is 12 times d , that is around 2 mm. This length is in the order of the diameter or perimeter of a needle.

Vibrations of the needle

Is it possible that once hit by the ultrasound wave, the needle starts vibrating and emitting a signal? Such a delayed signal will be interpreted by the probe as an object being behind the needle. This might explain the artifacts.

A way to test this hypothesis is to compute the sound pressure level (SPL) on a needle and then to obtain the resulting displacement D of the needle. To decide whether such displacement could produce an ultrasound, we have to compare D to the wavelength of the ultrasound. The 8658 rectal probe used by Nucletron emits and receives 5 to 7.5 MHz signals[†]. Since the velocity of sound in water and human tissues is around 1500 m/s, the corresponding wavelength is around $1.5 \cdot 10^3 / 5 \cdot 10^6 = 3 \cdot 10^{-4}$ m (300 microns). In section 5.2 we show that the pressure on the needle is completely below that range (by a factor 10^{15}). Hence the pressure on the needle is 10^{15} times weaker than what is required to produce artifacts. We must conclude that vibrations of the needle do not explain the artifacts.

Displacement of the needle

The sound pressure level (SPL) in dB is equal to $20 \cdot \log_{10}(\frac{p_{\text{rms}}}{p_{\text{ref}}})$ where $p_{\text{ref}} = 10^{-6}$ Pa in water and p_{rms} is the *root mean square* pressure. The 8658 rectal probe used by Nucletron has a 60 dB SPL which gives $p_{\text{rms}} = 10^{60/20} \cdot p_{\text{ref}} = 10^{-3}$ Pa. Let us consider a length l cylindrical section of the needle (of diameter $1.5 \cdot 10^{-3}$). The density of iron is 7860 kg/m³. Hence,

1. only the half part of the needle in front of the wave is submitted to the sound pressure (see [16]). Half of the outer area of the cylinder is $\pi \cdot l \cdot (1.5/2) \cdot 10^{-3} \text{ m}^2$. Hence the 10^{-3} Pa pressure produces a force of $F = \pi \cdot l \cdot (1.5/2) \cdot 10^{-6}$ N on the needle.

[†]<http://www.bkmed.com/applications/urology/brachytherapy.asp>

2. The mass of the cylinder is $M = 7860 \cdot \pi \cdot l \cdot (1.5/2)^2 \cdot 10^{-6}$ kg.
3. Let us assume that the position $X(t)$ of the needle is a harmonic vibration $X(t) = D \cdot \sin(\omega t)$ where $\omega = 2\pi f$ and the frequency f is similar to the frequency of the produced ultrasound (5 to 7.5 MHz). The force on the needle is $M \cdot X''(t)$ which has magnitude of $MD\omega^2$.
4. Combining 1 and 3 we obtain that the displacement of the needle is at most around $D = F/M\omega^2 = \frac{\pi \cdot l \cdot (1.5/2) \cdot 10^{-6}}{7860 \cdot \pi \cdot l \cdot (1.5/2)^2 \cdot 10^{-6} \cdot 4\pi^2 \cdot 25 \cdot 10^{12}} \approx 2 \cdot 10^{-19}$ m.

Therefore as this vibration displacement must produce an “artifact” ultrasound beam of wavelength around $3 \cdot 10^{-4}$ m, this fully contradicts the order of magnitude of the effective displacement D of the needle (a factor 10^{15} difference). As the size of an iron atom is generally estimated around $3 \cdot 10^{-10}$ m, the displacement of the needle is 10^9 times smaller than the iron atom. The emitted ultrasound has not any vibration effect on the needle.

The conclusion of this section must be that until now we do not have a plausible physical explanation for the artifacts. Therefore we move on to mathematical imaging techniques that at least can help to suppress the impact of the artifacts on the screen.

5.3 Medical imaging techniques

First we briefly explore the field of medical imaging, after that we explain the techniques we used for cleaning up the images.

A short impression and outlook of Mathematical Imaging

Mathematical imaging is a very active research area. Image restoration is the process of attempting to correct for degradation in a recorded image. Several types of problems can be distinguished: image denoising, image deblurring, and image inpainting. We briefly discuss each of these types.

Image denoising

In image denoising, the image is degraded by noise, for example during transmission. The model for linear noise is

$$g = f + n,$$

where the vector f represents the original image, n is the noise vector, and g is the vector of the degraded image. The goal of image denoising is the recovery of the original image f from the degraded image g . In many applications, a bound $\|n\|_2 \leq \nu$, or an estimate $\|n\|_2 \approx \nu$ for the noise is known, which one may try to exploit.

Image deblurring

In image deblurring, the continuous model is

$$g(s) = \int_D k(s, t) f(t) dt,$$

where D is a domain and k is an integral kernel. In operator form this equation is a Fredholm integral equation of the first kind

$$g = Kf,$$

where the inverse operator K^{-1} is unbounded. In the discrete case, we would like to find an approximation to the solution f given discrete, error contaminated data of the form

$$\tilde{g}_j = g_j + n_j,$$

where

$$g_j = g(s_j) = \int_D k(s_j, t) f(t) dt, \quad n_j = n(s_j),$$

and the error n is discrete white noise, which means that

$$E(n_j) = 0, \quad E(n_i n_j) = \begin{cases} \sigma^2 & i = j, \\ 0 & i \neq j, \end{cases}$$

where E denotes expectation and σ^2 is the variance. Typically, the noise is assumed to be Gaussian distributed, and spatially invariant. Solving for f given g is called an inverse problem. These problems are challenging since some regularization technique has to be used to prevent the errors n_j from blowing up in the solution; therefore this type of problems is called ill-posed (cf., [7]). The most popular choice is Tikhonov regularization, which approximates f by minimizing an expression of the form

$$\min_f \|Kf - g\|_2^2 + \lambda^2 \|Lf\|_2^2,$$

for a certain regularization parameter λ and regularization operator L . Often L is taken to be the identity, while many authors have studied sensible ways to choose λ . One of the alternatives for solving ill-posed systems is formed by the truncated singular value decomposition. Another regularization technique for images is the total variation method [17].

Image inpainting

In image inpainting part of an image is missing; see [1, 4, 3] for more information.

Further trends in image processing

We would like to mention a few recent trends in image restoration:

- methods that preserve the edges of the image ([17] and for instance [5]);
- (iterative) methods that preserve the nonnegativity of the pixel values (see [15, 6]);
- restoring images with spatially-variant blur (see [12, 13, 2]);
- the use of point spread functions [12, 18];
- deblurring using multiple images [18];
- the use of subimages [14];
- determining the statistical confidence we can have in the pixel values or features they form in the images [14].

Another relevant field for this project is image registration: determining a geometrical transformation that aligns points in one view of an object with corresponding points in another view of the same object, or a similar object (see [9, 11, 10]). Yet another subfield of imaging is image segmentation, see [8].

Cleaning up the image

We note that in our case the noise is periodic, and hence in particular spatially variant. In order to identify these periodic patterns Fourier analysis can be used. The Fast Fourier Transform (FFT) decomposes a signal into contribution of waves of particular frequency. The periodic patterns in the image should correspond to a peak in the Fourier transform at the frequency of the pattern. Removing this peak and inverting the Fourier transform will eliminate the patterns which occur at that frequency. The Fourier transform can also be used to remove noise from a signal. In signal processing, noise generally appears as high frequency contributions in the frequency domain. Removing these high frequency contributions produces a cleaned image.

Coordinate transformation

In order to identify the periodic artifacts we first convert the rectangular coordinate image into polar coordinates. The original image is given in Figure 5.7.

We crop the image to remove the text and focus only on the position of the needles. This results in the image in polar coordinates Figure 5.8

This represents the ultra sound images as measured by the probes. The artifacts now lie in a vertical line behind the needle positions. This change of coordinates allows us to take the Fourier transform in the vertical direction only rather than using a two dimensional transform.

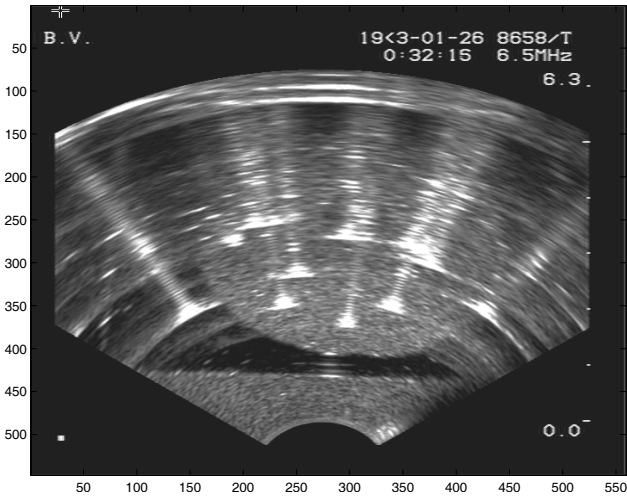


Figure 5.7: The ultra sound image in rectangular coordinates

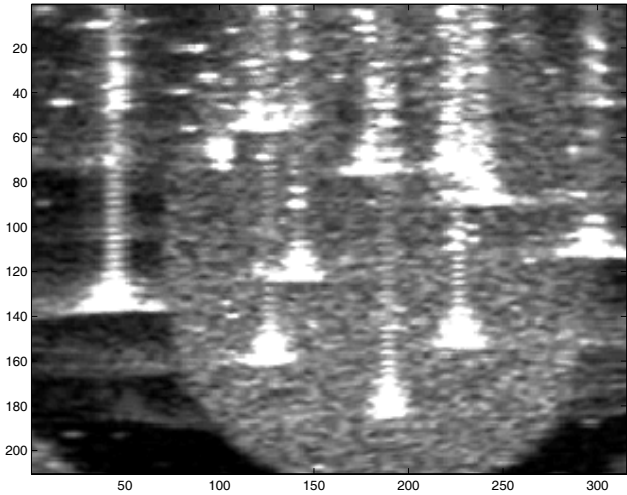


Figure 5.8: The ultra sound image in polar coordinates

Fourier transform

Importing the image into Matlab converts the image into a matrix of real valued scalars. The image can now be considered as a series of vertical lines each represented by a vector. The one dimensional discrete FFT is applied to each vector. In the frequency domain we remove all signals above a fixed frequency. This frequency chosen to be large enough to retain the important information in the image, but sufficiently small to remove the periodic artifacts

and denoise the image. The inverse Fourier transform is applied to give the filtered image Figure 5.9.

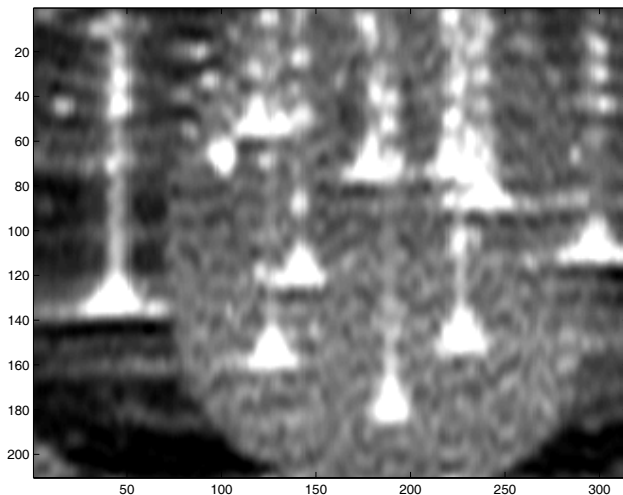


Figure 5.9: The *cleaned* ultra sound image after Fourier analysis

Thresholding

Finally thresholding can then be applied to the filtered image to aid in identifying the needle positions. Taking a threshold makes a black and white picture from a grayscale picture by making all pixels above a certain value white and all others black. If this value is chosen well, the needles should show up as white blobs in the picture, see Figure 5.10.

Acknowledgments We thank Stefan Henn (Heinrich-Heine-Universität Düsseldorf) for his useful comments on the section about medical imaging, Sjoerd Rienstra (Technische Universiteit Eindhoven) for sharing his ideas on the energy argument and Jeroen Schuurmans from Nucletron for bringing us this problem, the time he took to explain all the details to us and his enthusiasm. We also thank Jan Bouwe van den Berg (Vrije Universiteit Amsterdam), Remco Duits (Technische Universiteit Eindhoven), Robbert Fokkink (Technische Universiteit Delft), Erik Franken (Technische Universiteit Eindhoven) and Vivi Rottschäfer (Universiteit Leiden) for their input during the SWI.

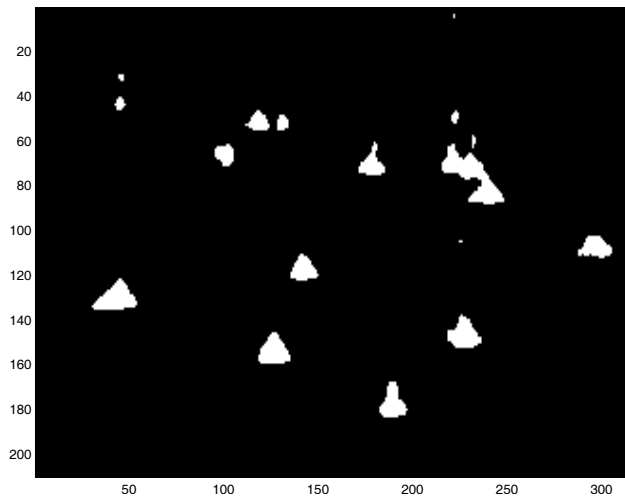


Figure 5.10: The needle positions highlighted using thresholding on the *cleaned* image

5.4 Bibliography

- [1] M. BERTALMIO, G. SAPIRO, V. CASELLES, AND C. BALLESTER, *Image inpainting*, in ACM Siggraph, Computer Graphics Proceedings, K. Akeley, ed., 2000, pp. 417–424.
- [2] D. CALVETTI, B. LEWIS, AND L. REICHEL, *Restoration of images with spatially variant blur by the GMRES method*, in Advanced Signal Processing Algorithms, Architectures, and Implementations X; Proc. SPIE Conference, F. T. Luk, ed., vol. 4116, November 2000, pp. 364–374.
- [3] T. F. CHAN, S. H. KANG, AND J. SHEN, *Euler’s elastica and curvature-based inpainting*, SIAM J. Appl. Math., 63 (2002), pp. 564–592.
- [4] T. F. CHAN AND J. SHEN, *Mathematical models for local nontexture inpaintings*, SIAM J. Appl. Math., 62 (2001/02), pp. 1019–1043.
- [5] C. FROHN-SCHAUF, S. HENN, AND K. WITSCH, *Nonlinear multigrid methods for total variation image denoising*, Computing and Visualization in Science, 7 (2004), pp. 199–206.
- [6] M. HANKE, J. G. NAGY, AND C. VOGEL, *Quasi-Newton approach to nonnegative image restorations*, Linear Algebra Appl., 316 (2000), pp. 223–236.
- [7] P. C. HANSEN, *Rank-deficient and discrete ill-posed problems*, SIAM Monographs on Mathematical Modeling and Computation, Society for

- Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Numerical aspects of linear inversion.
- [8] R. M. HARALICK AND L. G. SHAPIRO, *Image segmentation techniques*, Computer Vision, Graphics, and Image Processing, 29 (1985), pp. 100–132.
- [9] S. HENN AND K. WITSCH, *Iterative multigrid regularization techniques for image matching*, SIAM J. Sci. Comput., 23(4) (2001), pp. 1077–1093.
- [10] ———, *Image registration based on multiscale energy information*, SIAM Journal on Multiscale Modeling and Simulation (MMS), 4(2) (2005), pp. 584–609.
- [11] J. MODERSITZKI, *Numerical methods for image registration*, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, 2004. Oxford Science Publications.
- [12] J. G. NAGY AND D. P. O’LEARY, *Fast iterative image restoration with a spatially-varying PSF*, in Advanced Signal Processing: Algorithms, Architectures, and Implementations VII; Proc. SPIE Conference, F. T. Luk, ed., vol. 3162, October 1997, pp. 388–399.
- [13] ———, *Restoring images degraded by spatially variant blur*, SIAM J. Sci. Comput., 19 (1998), pp. 1063–1082.
- [14] ———, *Image restoration through subimages and confidence images*, Electron. Trans. Numer. Anal., 13 (2002), pp. 22–37.
- [15] J. G. NAGY AND Z. STRAKOS, *Enforcing nonnegativity in image reconstruction algorithms*, in Mathematical Modeling, Estimation, and Imaging; Proc. SPIE Conference, D. C. Wilson, H. D. Tagare, F. L. Bookstein, F. J. Preteux, and E. R. Dougherty, eds., vol. 4121, October 2000, pp. 182–190.
- [16] S. W. RIENSTRA AND A. HIRSCHBERG, *Elements of aero-acoustics*, Lecture Notes of Von Karman Lecture Series, (1994).
- [17] L. I. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Physica D, 60 (1992), pp. 259–268.
- [18] R. VIO, J. G. NAGY, AND W. WAMSTEKER, *Multiple-image composition and deblurring with spatially-variant PSFs*, Astronomy & Astrophysics, 434 (2005), pp. 795–800.

MATH SAVES THE FOREST

Analysis and optimization of message delivery in wireless sensor networks

Peter Korteweg¹, Misja Nuyens², Rob Bisseling³, Tom Coenen⁴,
Henri van den Esker⁵, Bart Frenk¹, Roland de Haan⁴,
Birgit Heydenreich⁶, Remco van der Hofstad^{1,7}, Jos in 't Panhuis¹,
Lieneke Spanjers⁸, Maarten van Wieren⁷

Abstract

Wireless sensor networks are decentralised networks consisting of sensors that can detect events and transmit data to neighbouring sensors. Ideally, this data is eventually gathered in a central base station. Wireless sensor networks have many possible applications. For example, they can be used to detect gas leaks in houses or fires in a forest.

In this report, we study data gathering in wireless sensor networks with the objective of minimising the time to send event data to the base station. We focus on sensors with a limited cache and take into account both node and transmission failures. We present two cache strategies and analyse the performance of these strategies for specific networks. For the case without node failures we give the expected arrival time of event data at the base station for both a line and a 2D grid network. For the case with node failures we study the expected arrival time on two-dimensional networks through simulation, as well as the influence of the broadcast range.

KEYWORDS: sensor networks, data gathering, stochastic optimisation, distributed algorithms, random walks, first-passage percolation.

6.1 Introduction

Suppose that you want to design a system to detect fires in a forest. You consider placing sensors that can detect a fire in their neighbourhood. Since

1: Technische Universiteit Eindhoven, 2: Vrije Universiteit Amsterdam, 3: Universiteit Utrecht, 4: Universiteit Twente, 5: Technische Universiteit Delft, 6: Universiteit Maastricht, 7: EURANDOM, 8: CQM

these sensors work on battery power, immediately restrictions arise that make the design of such a system an interesting endeavour. First, transmitting a message to a receiver outside the forest may cost too much energy. In that case, only short-range transmissions are possible. Also, it may not be feasible or too costly to replace the battery of the sensors on a regular basis, so the battery lifetime should be made as long as possible. On the other hand, you want to be sure that the message that there is a fire is transmitted to the receiver outside the forest, and moreover, this should not take too much time.

The question how to design such a forest fire detection system, and control the efficiency of such a system in terms of observing a fire at the base station given possible sensor failures, is an example of the question posed to SWI 2006 by Chess [1]. Chess is a middle-size company providing products and services in the field of electronics, IT-applications, and embedded software. At the moment, Chess considers designing so-called wireless sensor networks for a broad range of applications. We shall describe those networks in more detail further on in this introduction. Apart from detecting forest fires, one could think of detecting gas leaks in neighbourhoods, monitoring the functioning of street lights, as well as using the system for picking up garbage: in many Dutch cities, garbage is collected in large underground bins, and these bins could send a message when they are (almost) full and need to be emptied.

A wireless sensor network is a network that consists of small devices that communicate with each other through radio signals. See Figure 6.1 for an example of such a sensor. Such devices, named sensor nodes, are able to monitor their environment, collect environmental data, process these data and communicate them to other nodes [6, 7].

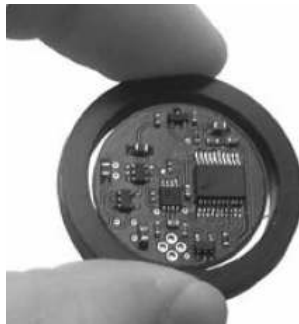


Figure 6.1: A wireless sensor

Sensor networks have several characteristics that distinguish them from wired networks, see [9]. We list the differences with an emphasis on those differences that influence the design of communication algorithms:

- Sensors mainly use *broadcasting* to communicate data. A sensor node that broadcasts, sends data via a radio signal to all sensors in its neighbourhood.

- Sensor networks are *distributed* networks, i.e., they lack a central coordinator. As a consequence, each node has to decide itself what it communicates and when.
- Sensor nodes are *limited in memory and power*.
- Sensor *nodes* are prone to *failure*, i.e., a node may break down and stop operating.
- Wireless *communication* is prone to *failure*.

One of the key research problems in the area of sensor networks is finding efficient communication algorithms. Much research has focused on finding such algorithms for wired networks; see for example the surveys [3] and [5]. However, due to their characteristics, communication algorithms for wired networks do not necessarily provide algorithms for wireless sensor networks. Therefore, in recent years research focused on finding efficient communication algorithms for wireless sensor networks; see [9] for an overview of such algorithms.

In this report, we study a communication problem on a static wireless sensor network, the SENSOR DATA GATHERING PROBLEM (SDGP). In this problem, stations (sensor nodes) in the network provide data that need to be gathered at a base station; the stations are assumed to be static. These data consist of events that occurred in the neighbourhood of a node, e.g. a fire. Stations may communicate messages of events through broadcasting and each message contains information concerning a single event. The objective of the SDGP is to find an efficient algorithm for data gathering at a base station of a wireless static sensor network. Data gathering means that for each event at least one message containing the data should reach the base station. In the literature, there exist several concepts of efficiency. These concepts focus on minimising a function of the completion time of data gathering or maximising a function of the battery lifetime. In this paper, we mainly focus on the objective of minimising the completion time. So, generally speaking the objective is to send messages to the base station as fast as possible.

We emphasise three specific characteristics of our sensor network. First, the network is prone to two types of failure: communication failure and node failure. Second, nodes have a limited memory to store messages, called the *cache*. Due to their limited cache size, sensors should have a *cache strategy*, which determines which message to delete in case of a cache overflow. Third, for design purposes and to limit battery power, sensors are simple devices with a limited set of operations. To communicate their data, sensors use broadcasting. Thus, we assume that sensors cannot use any specific routing information, i.e., sensors are unable to establish point-to-point communication of messages.

Summarising, a communication algorithm should consist of a protocol that decides which messages to broadcast, and of a cache strategy. In this paper, we analyse the performance of several communication algorithms for specific network structures: the 1D grid, the 2D (square) grid and the 2D hexagonal

grid. Since the charm and power of the described sensor networks lies in their simplicity, we focus on communication algorithms that are as simple as possible.

The paper is organised as follows: in Section 6.2, we give a mathematical formulation of the problem. In Section 6.3, we give a mathematical analysis for the case without node failures and unit broadcast radius. First, we analyse the SDGP with unlimited cache size. In this case, there is no need for a cache strategy and message detection by the base station is independent of other messages. We give a probabilistic analysis of the expected number of rounds before an event is detected by the base station. Then, we analyse the SDGP with a cache size of one. In this case, events cannot always be detected by the base station. We give a probabilistic analysis for the case with two events. In Section 6.4, we consider the more general case with node failures and arbitrary broadcast radius. Our results in this section are based on simulations only. In Section 6.5, we summarise the results and give recommendations for designing efficient communication algorithms.

6.2 Problem formulation

We formulate the SENSOR DATA GATHERING PROBLEM as a graph problem. Let $G = (V, A)$ be a directed graph with vertex set V , edge set A , and let $|V| = n$. Also given are a *sink* $s \in V$, a set of events $E = \{1, \dots, |E|\}$, a set of messages $M = \{1, \dots, m\}$ for some integer m , an integer cache size $c > 0$, an integer broadcast radius $r > 0$ and probabilities $p > 0$ and $q > 0$, defined below.

The nodes of the graph are stations and the sink is the base station. For each pair of nodes $u, v \in V$, we define the *distance* between u and v , denoted by $d(u, v)$, as the edge cardinality of a shortest path from u to v in G . Given radius r let $N_r(u) = \{v | d(u, v) \leq r\}$ be the neighbourhood of u and let $v \in N_r(u)$ be a neighbour of u . In case $r = 1$, the neighbours of u are those nodes v such that $(u, v) \in A$. We assume that time is discrete, say $\{1, 2, \dots\}$; a time instance is called a *round*. We assume that sensor nodes have a clock, and that all clocks are synchronised.

Each event $e \in E$ contains data, e.g. “There is a fire”, a *source node* v_e , i.e., the node where the event was detected, and a *detection time* t_e , i.e. the first time the source node detected the event. Nodes may communicate with each other and if they communicate, they exchange messages. Each message $j \in M$ contains data of a single event e , including source node and detection time. It also contains a timestamp, indicating the round in which the message was sent by its source node. Nodes may use this information to schedule messages. We assume that once the source node of event e detects this event, it creates a message for this event in *each* subsequent round. Note that these messages all have the same detection time, but different timestamps.

Each node has a cache to store messages. We assume the cache consists of a receiver cache of unlimited size and a sender cache of size c . During a round, nodes may communicate with each other through *broadcasting*. A node that

broadcasts sends a copy of each message in its sender cache to all its neighbours. So, if node u broadcasts its sender cache, then each node $v \in N_r(u)$ receives the content of the sender cache of u and stores the information in its receiver cache. A node may broadcast at most once during a round. We assume that node broadcasts do not interfere with each other, hence there is no collision of messages.

A sensor network is prone to two types of failure: communication failure and node failure. We define q as the probability that a broadcast from node u to v during a round is a success, for any $(u, v) \in A$. Moreover, we assume that broadcasts fail independently. In particular, this means that if in a round node u broadcasts to both v and v' , then each node has probability q to receive (the same) data.

Since the time scale for the transmission of a message through the network is of a different order than the lifetime of a node, we assume that nodes do not fail during the period that we consider. We call a node *active* if it is operational, i.e., it has not failed, and *inactive* if it has failed. We define p as the probability that a node is active, and assume that nodes are active independently of each other.

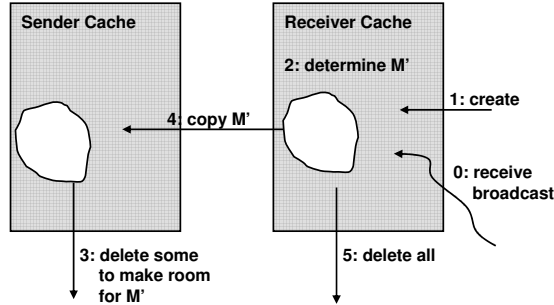


Figure 6.2: The cache strategy

Each node v has a cache strategy. We assume that at the start of a round, all messages received in the previous round are stored in the receiver cache. A node should then update its cache using its cache strategy. A cache update of node v consists of the following consecutive actions, see also Figure 6.2.

1. Create a message for each event with source node v and store this message in the receiver cache.
2. Choose the set of messages M' to be copied from the receiver cache to the sender cache; below we will consider two ways to choose this set.
3. Delete messages from the sender cache such that all messages in M' can be copied to the sender cache.

4. Copy the messages in M' to the sender cache.
5. Delete *all* messages from the receiver cache.

We make the following two assumptions for each cache update. First, after the cache update the sender cache contains at most one message for each event e . If the cache consists of multiple messages for a single event before the update, then the cache only stores the message with the most recent timestamp. Second, messages are only deleted from the sender cache when necessary.

In case of a limited sender cache c , the node must choose which messages to delete from the sender cache and which messages to copy from the receiver cache to the sender cache. The cache strategy should be based only on local information of a node: the current time and the information of the messages in its cache. Hence, the cache strategy is a distributed algorithm.

We consider two different cache strategies based on how messages are deleted from the sender cache in step 3:

- **RANDOM DELETION:** Messages are randomly deleted from the sender cache;
- **TIMESTAMP DELETION:** Messages are deleted by decreasing timestamp, i.e., the message with the oldest timestamp is deleted first. Ties are broken arbitrarily, i.e., if two messages have the same timestamp, one of them is deleted according to some arbitrary but fixed rule.

The cache strategy **TIMESTAMP DELETION** was introduced by Chess [1]. In fact, Chess' cache strategy also deletes too old messages if the cache is not full. However, this does not make sense here since we do not consider optimising the battery lifetime. In Subsection 6.3, we further comment on this when discussing the 2D grid. Furthermore, note that under strategy **RANDOM DELETION** it is possible that a node has detected an event, but it does not send a message of this event immediately. Finally, if one or more messages from an event reach the base station, we say that the event (data) has been *gathered* by the base station.

The objective of the **SENSOR DATA GATHERING PROBLEM** is to gather events at the base station of a wireless static sensor network while minimising the completion time of all events. The completion time of an event is the number of rounds needed for one of the messages corresponding to this event to reach the base station. Thus, generally speaking, we are interested in sending messages to the sink as fast as possible. Since the completion time of an event depends on the probabilities p and q , it is a random variable.

6.3 Probabilistic analysis

In this section, we give a probabilistic analysis for the case without node failures, i.e., $p = 1$ throughout this section. In Subsections 6.3 and 6.3, we consider

unlimited cache size ($c = \infty$) for both the 1D grid and the 2D grid. We are interested in the expected completion time of an event, i.e., the expected number of rounds needed to send some message with data of event e to the sink. As the cache size is infinite, no messages have to be deleted from the sender cache. Hence, the completion time of an event is independent of the possible existence of other messages. Thus we may restrict our analysis to considering detection of a single event. Another consequence is that the cache strategies RANDOM DELETION and TIMESTAMP DELETION are identical. Let the random variable T_d be the number of rounds required to gather at the sink a message whose source node is at distance d from the sink.

In the Subsections 6.3 and 6.3, we consider a cache of size one and give a probabilistic analysis in case two events occur.

The 1D grid with unlimited cache size

Given is a 1D grid of sensor nodes $s, 1, \dots, n$ with base station s ; node i is at distance i from s , see Figure 6.3.

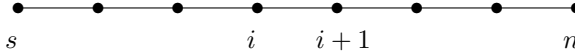


Figure 6.3: 1D grid with base station s

Suppose an event occurs at time 0 at node d . In each round, let X be the node closest to the sink that has received a message of this event. We will also call X the distance of the event to the sink. First, we consider the case $r = 1$, so nodes can only broadcast their cache to their nearest neighbours. In each round, X either moves one step closer to the sink, with probability q , or it remains at the same distance, with probability $1 - q$. In total, X has to travel distance d . This means that T_d is equal to the number of trials needed to obtain d successes, where the probability of a success is q . So, T_d follows a negative binomial distribution with parameters q and d , i.e.,

$$\mathbb{P}(T_d = t) = \binom{t-1}{d-1} q^d (1-q)^{t-d}, \quad t = d, d+1, d+2, \dots \quad (6.1)$$

Note that as a consequence, for a broadcast success probability $q > 0$, the event will be gathered with probability 1. Another consequence of (6.1) is the following.

Corollary 6.3.1. *The expected number of rounds required to gather an event detected at distance d satisfies $\mathbb{E}[T_d] = d/q$.*

Second, consider the situation that a node is able to transmit at a larger range, i.e., $r > 1$. We assume that the success probability of a broadcast

equals q independent of the distance between the nodes. Let Y_r be the effective distance that one particular message gets closer to s in an arbitrary round. The effective distance is the maximum distance over which the communication is successful; hence, Y_r is a random variable.

As the success probability of communication is independent of the radius, the probability to get r steps closer to the sink is q . Similarly, given that this broadcast fails, then the probability to get $r - 1$ steps closer to the sink is q . Continuing this argument, we arrive at:

$$\begin{aligned}\mathbb{P}(Y_r = k) &= (1 - q)^{r-k}q, & k = 1, 2, \dots, r, \\ \mathbb{P}(Y_r = 0) &= (1 - q)^r.\end{aligned}\tag{6.2}$$

Using (6.2), we are able to find the expected value of Y_r . First we write

$$\begin{aligned}\mathbb{E}[Y_r] &= \sum_{k=0}^r k\mathbb{P}(Y_r = k) = \sum_{k=1}^r k(1 - q)^{r-k}q = q \sum_{i=0}^{r-1} (r - i)(1 - q)^i \\ &= qr \frac{1 - (1 - q)^r}{1 - (1 - q)} - q(1 - q) \sum_{i=1}^{r-1} i(1 - q)^{i-1}.\end{aligned}\tag{6.3}$$

To evaluate the sum in (6.2), we write

$$\begin{aligned}\sum_{i=1}^{r-1} i(1 - q)^{i-1} &= -\frac{d}{dq} \sum_{i=1}^{r-1} (1 - q)^i = -\frac{d}{dq} \frac{(1 - q) - (1 - q)^r}{q} \\ &= \frac{1}{q^2} - \frac{(1 - q)^r}{q^2} - \frac{r(1 - q)^{r-1}}{q}.\end{aligned}$$

Plugging this into (6.3), we get

$$\begin{aligned}\mathbb{E}[Y_r] &= r - r(1 - q)^r - \frac{1 - q}{q} + \frac{(1 - q)^{r+1}}{q} + r(1 - q)^r \\ &= r + 1 + \frac{(1 - q)^{r+1} - 1}{q}.\end{aligned}$$

Note that for $r = 1$, we find $\mathbb{E}[Y_1] = q$, which corresponds to Corollary 6.3.1.

However, if $r > 1$, then a message can be overtaken by messages with a later timestamp. Hence, in this case, Y_r is a lower bound on the effective distance that one particular message gets closer to s in an arbitrary round.

In Figure 6.4, we have plotted $\mathbb{E}[Y_r]$ as a function of q for several choices of r . The figure confirms what is intuitively obvious: for broadcast radius $r > 1$, the effective number of steps is much larger than in case $r = 1$. In fact, for $r = 1$ the curve is linear, but for $r > 1$ the curves are larger than the linear curves $f(q) = rq$. The reason for this is that if broadcast to a node at distance r fails, there is still a positive probability that a broadcast to a node of distance less than r is successful, or that a younger message is overtaking the message closest to the base station.

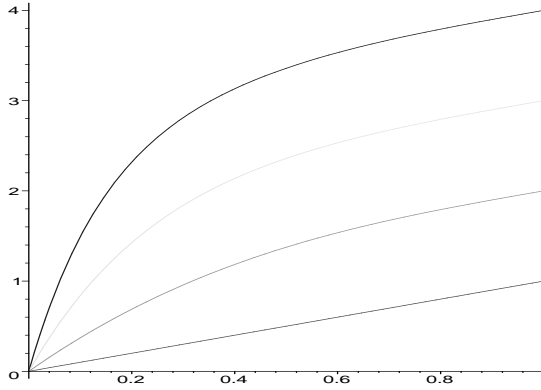


Figure 6.4: $\mathbb{E}[Y_r]$ as a function of q for $r = 1, 2, 3, 4$ (from bottom to top)

The 2D grid with unlimited cache size

Given is a 2-dimensional finite grid of size $\sqrt{n} \times \sqrt{n}$ for some integer \sqrt{n} and with base station s located at one of the corners. Note that on the grid the distance of a node from the base station is at most $2\sqrt{n}$, see Figure 6.5.

Since the probability that an event is gathered is equal to 1 on a 1D grid, as we have seen in the previous section, it is also equal to 1 on a 2D grid. Therefore, we turn to the analysis of $\mathbb{E}[T_d]$, the expected time that is needed to gather an event whose source node, say v_d , is at distance d from the sink. Let X_a be a random variable indicating the number of rounds needed in order to communicate successfully via the (directed) edge a . Clearly, the variables X_a are independently and identically distributed following a geometric distribution with success probability q for all edges a . For any path Φ that connects the sensor to the base station, let T_Φ be the random variable that indicates the time needed to successfully communicate the event via path Φ . Finally, let ϕ be a shortest path from node v_d to the base station, where $a \in \phi$ means that edge a is part of the path ϕ . Then

$$\mathbb{E}[T_d] = \mathbb{E}[\min_{\Phi} T_\Phi] \leq \min_{\Phi} \mathbb{E}[T_\Phi] = \mathbb{E}[T_\phi] = \mathbb{E}[\sum_{a \in \phi} X_a] = \sum_{a \in \phi} \mathbb{E}[X_a] = \frac{d}{q}.$$

Note that the upper bound d/q corresponds with Corollary 6.3.1 (when $r = 1$).

The remainder of this section is devoted to so-called first-passage percolation. The theory of first-passage percolation examines how T_d behaves depending on the position of the sensor with respect to the base station. That is, it examines the behaviour of sensors dependent on whether they are for example situated on the same grid-line as the base station or whether their position is diagonal with respect to the base station. Note that this position influences the number of shortest paths over which a message can be communicated from its source node to the base station.

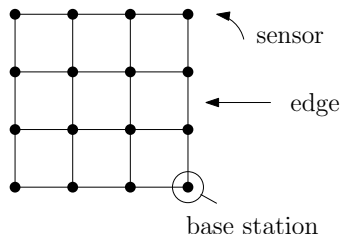


Figure 6.5: The 2D grid

First-passage percolation with geometric distributions has attracted much attention in the literature (see e.g. [4, 8]). An important result, the so-called shape-theorem, describes the shape of the set of points on the grid that can be reached through communication starting from a fixed source sensor within a certain time. The following theorem is implied by the shape-theorem and we use techniques from first-passage percolation to prove it. The theorem provides an upper bound on the probability that the time needed to communicate the event via a specific shortest path ϕ of length d is εd more than the expected time d/q for some positive constant ε .

Theorem 6.3.2. *The probability that T_ϕ exceeds its expectation d/q decreases exponentially with the excess time:*

$$\mathbb{P}(T_d \geq \frac{d}{q} + \varepsilon d) \leq e^{-d \frac{\varepsilon^2 q^2}{2(1-q)}}.$$

Proof. Consider $\mathbb{P}(T_d \geq dk)$ for some positive constant k and let ϕ be a shortest path from node v_d to the base station. Then for all $t \geq 0$ the following is true:

$$\begin{aligned} \mathbb{P}(T_d \geq dk) &\leq \mathbb{P}(T_\phi \geq dk) = \mathbb{P}\left(\sum_{a \in \phi} X_a \geq dk\right) \\ &= \mathbb{P}(e^{t \sum_{a \in \phi} X_a} \geq e^{tdk}) \leq e^{-tdk} \mathbb{E}[e^{t \sum_{a \in \phi} X_a}]. \end{aligned}$$

The last inequality follows from applying the Markov inequality. Since this holds for all $t \geq 0$, we have

$$\mathbb{P}(T_d \geq dk) \leq \min_{t \geq 0} \exp(-tdk + \log \mathbb{E}[e^{t \sum X_a}]) = \min_{t \geq 0} (\exp(-tk + \log \mathbb{E}[e^{tX_1}]))^d,$$

where X_1 is equal to one of the random variables X_a for some edge a on the path ϕ . Hence, we can write

$$\mathbb{P}(T_d \geq dk) \leq e^{-dI(k)},$$

where

$$\begin{aligned} I(k) &= \sup_{t \geq 0} \{kt - \log \mathbb{E}[e^{tX_1}]\} = \sup_{t \geq 0} \{kt - \log \frac{qe^t}{1 - (1-q)e^t}\} \\ &= \sup_{t \geq 0} \{(k-1)t - \log q + \log(1 - (1-q)e^t)\}. \end{aligned}$$

Calculus yields

$$\begin{aligned} I(k) &= (k-1)(\log(k-1) - \log k - \log(1-q)) - \log q + \log \frac{1}{k} \\ &= (k-1) \log\left(\frac{k-1}{1-q}\right) - k \log k - \log q. \end{aligned}$$

Setting $k := \frac{1}{q} + \varepsilon$, we get:

$$\mathbb{P}(T_d \geq d(\frac{1}{q} + \varepsilon)) \leq e^{-dI(\frac{1}{q} + \varepsilon)}.$$

Since $I(\frac{1}{q}) = I'(\frac{1}{q}) = 0$, using the Taylor expansion yields

$$I(\frac{1}{q} + \varepsilon) = I(\frac{1}{q}) + \varepsilon I'(\frac{1}{q}) + \frac{\varepsilon^2}{2} I''(\frac{1}{q}) + o(\varepsilon^2) = \frac{\varepsilon^2}{2} I''(\frac{1}{q}) + o(\varepsilon^2).$$

Finally, calculating

$$I''(\frac{1}{q}) = \frac{q^2}{1-q},$$

completes the proof. \square

To illustrate Theorem 6.3.2, Figure 6.6 shows this upper bound for the situation where $\sqrt{n} = 101$, $d = 200$ (the worst case scenario) and $q = 0.95$.

It is clear that for the expected number of steps needed, which is around 210, the upper bound does not provide much information. However, for the situation with only ten steps more, Theorem 6.3.2 provides strong information: the probability that the message needs more than 220 steps to reach the base station is already below 1%.

Let us emphasise that the given bound only takes the communication via one path into account. In reality, there are multiple paths that can be used, hence it is likely that the expected completion time will be even shorter.

For the example shown, we can also conclude that if the cache strategy would delete messages whose timestamp is at least 220 rounds old then the probability that the first message created for this event does not reach the base station is less than 1 percent. Such a strategy would assure that messages are not kept longer than necessary in the cache, and decreases the amount of old messages circulating in the network. This is beneficial for the lifetime of the batteries in the sensor.

We conclude this subsection with the following remark. In this subsection, we have assumed that sensors do not fail, i.e., $p = 1$. This assumption is not

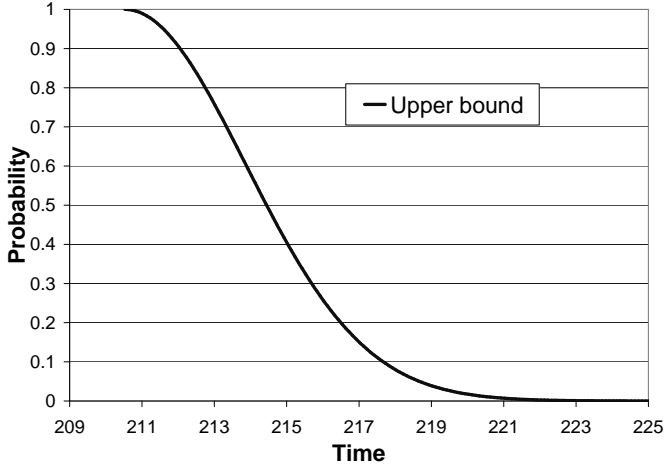


Figure 6.6: The upper bound of Theorem 6.3.2 for $\sqrt{n} = 101, d = 200$ and $q = 0.95$.

very restrictive in the 2-dimensional case, as from an arbitrary sensor in the network, there exist multiple paths toward the base station. So, if one path is not available, there may be many other available candidates. Hence, unlike in the one dimensional case, for p not too far from 1, the probability that the base station can be reached from the sensor by at least one path is close to 1 as well. Of course, it would be an interesting problem to quantify these “not too far from 1” and “close to 1”.

The 1D grid with cache size one and two events

We again consider the model on the 1D grid, but this time we assume that $c = 1$ for each node. Given is a 1D grid of sensor nodes $s, 1, \dots, n$ with base station s ; node i is at distance i from s . We assume that the broadcast radius r is 1. We are interested in the probability that if two events occur, both events are gathered. We compare this probability for the cache strategies RANDOM DELETION and TIMESTAMP DELETION.

We assume there are two events 1 and 2 with source nodes v_1 and v_2 , respectively, and detection times t_1 and t_2 . Without loss of generality we assume that v_1 is closer to the sink than v_2 . Figure 6.7 illustrates the situation. First, we consider TIMESTAMP DELETION. In this case, from all the messages that a node receives, it will only send the one with the youngest timestamp, i.e., the message that has been in the system the shortest time. Let A_i be the event that the i th event is gathered. We are interested in calculating the

probabilities $\mathbb{P}(A_1)$ and $\mathbb{P}(A_2)$.

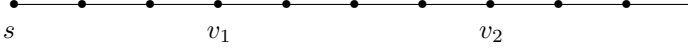


Figure 6.7: 1D grid with events 1 and 2

The probability $\mathbb{P}(A_1)$ is easily determined. Since messages corresponding to event 1 are sent every round after event 1 has been detected, there will always be a message from this event in the sender cache of v_1 . Hence, the probability that a single message reaches the sink is at least q^{v_1} . This implies that the probability that none of the messages sent by v_1 reach the sink is upper bounded by $\lim_{n \rightarrow \infty} (1 - q^{v_1})^n = 0$. Hence, $\mathbb{P}(A_1) = 1$.

Now we consider $\mathbb{P}(A_2)$. We claim that if $t_1 \leq t_2$, then $\mathbb{P}(A_2) = 0$. Indeed, any message created at v_2 must eventually pass vertex v_1 . However, since $t_1 \leq t_2$, this vertex is already busy sending messages of its own event. As the timestamp of these messages is always younger than the timestamp of messages from event 2, messages from event 2 can never pass v_1 , and thus never reach the sink. We conclude that if we want both events to be gathered at the sink, then the cache strategy with **TIMESTAMP DELETION** is bad one.

Next, suppose that in **TIMESTAMP DELETION** we have $t_2 < t_1$. For each round, let the random variable X be the position of the message from event 2 that is closest to the sink at time t_1 . Set for notational convenience $\tau = t_1 - t_2$. If $0 < k \leq v_2$, then $\mathbb{P}(X = k)$ is the probability of $v_2 - k$ successes in τ trials with success probability q . Hence, X is binomially distributed with parameters τ and q . Furthermore, $\mathbb{P}(X = 0)$ is the probability of at least v_2 successes in τ trials. Thus,

$$\mathbb{P}(X = k) = \binom{\tau}{v_2 - k} q^{v_2 - k} (1 - q)^{\tau - v_2 + k}, \quad \text{for } 0 < k \leq v_2, \quad (6.4)$$

$$\mathbb{P}(X = 0) = 1 - \sum_{i=0}^{v_2-1} \binom{\tau}{i} q^i (1 - q)^{\tau - i}. \quad (6.5)$$

Now we condition on the value of X . Since the strategy with **TIMESTAMP DELETION** implies that $\mathbb{P}(A_2 \mid X = k) = 0$ for all $k \geq v_1$, we have

$$\begin{aligned} \mathbb{P}(A_2) &= \sum_{k=0}^{\infty} \mathbb{P}(A_2 \mid X = k) \mathbb{P}(X = k) \\ &= \sum_{k=0}^{v_1-1} \mathbb{P}(A_2 \mid X = k) \mathbb{P}(X = k). \end{aligned} \quad (6.6)$$

To find these conditional probabilities, we consider the following situation. Let i and j be the position closest to the sink of the messages from v_1 and v_2

respectively. Let $p(i, j)$ be the probability that from this situation a message from v_2 reaches the sink. One round later, these positions are $i - 1$ and $j - 1$ with probability q^2 , and in that case the desired probability is $p(i - 1, j - 1)$. By also considering the other possibilities for the situation one round later, we find that $p(i, j)$ satisfies the recurrence relation:

$$p(i, j) = q^2 p(i - 1, j - 1) + q(1 - q)p(i, j - 1) + q(1 - q)p(i - 1, j) + (1 - q)^2 p(i, j). \quad (6.7)$$

Since messages from v_1 have priority over those from v_2 , the boundary conditions are $p(i, j) = 0$ if $i \leq j$, and $p(i, 0) = 1$ for all $i > 0$. Finally, observe that $\mathbb{P}(A_2 \mid X = k) = p(v_1, k)$. Hence, we can calculate (6.6) by solving the recurrence relation (6.7). Unfortunately, there is no easy closed-form solution of this recurrence relation, so that it is only useful for numerical purposes. In this report, we will not explore this numerical path.

Now we consider RANDOM DELETION. To simplify the analysis, we assume that $q = 1$. The case $q < 1$ will be studied via simulations in Section 6.4. If $v_2 - v_1 > t_1 - t_2$, then $\mathbb{P}(A_1) = 1$. We are therefore interested in finding $\mathbb{P}(A_2)$. Let t be the first round such that the sender caches of two adjacent nodes contain different messages. This situation is illustrated in Figure 6.8; here messages from v_1 are denoted by a circle, and those from v_2 by a square.

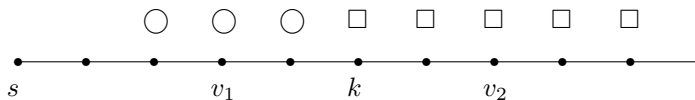


Figure 6.8: The first round that some node k contains a message of event 2 and node $k - 1$ contains a message of event 1.

Observe that the model from time t onward resembles a symmetric random walk (RW). Indeed, as a consequence of the RANDOM DELETION cache strategy, one round later, both nodes $k - 1$ and k are a circle, or a square, both with probability $1/2$, see also Table 6.1. So, with probability $1/4$, the front of the squares moves forward to $k - 1$, with probability $1/4$ it stays in k , and with probability $1/4$ it moves back to $k + 1$. The only difference with the RW model is the fourth option: a square moves to $k - 1$, and a circle moves to k . Although we have not managed to prove it, this fourth option seems to be no worse than the situation in which the front stays at k .

We may conclude that the front of squares reaches the sink in a time comparable to that of a RW with step size distribution

$$\mathbb{P}(X = 1) = \mathbb{P}(X = -1) = \mathbb{P}(X = 0)/2 = 1/4. \quad (6.8)$$

time t		time $t + 1$		
$k - 1$	k	$k - 1$	k	situation
○	□	○	□	non-moving front
		○	○	progressing front for message 1
		□	□	progressing front for message 2
		□	○	mixing front

Table 6.1: The four options for the evolution of the front.

So, results for this RW may give us (upper) bounds for the behaviour of the stream of squares. To describe the behaviour of this RW, we first quote two propositions about simple symmetric RW's, i.e., RW's that move one to the right or one to the left, both with probability $1/2$. These can be found in Chapter XIV of [2].

Proposition 6.3.3. *Consider a simple symmetric RW. Let $p(x, n)$ denote the probability that starting at $x \in \{s, 1, \dots, n\}$, the message reaches s before it reaches n . Then $p(x, n) = 1 - x/n$.*

This proposition is also known as the Gambler's ruin probability. Since the step size of our RW may be 0, it is not a simple symmetric RW. However, since the step-size distribution is symmetric and concentrated on $\{-1, 0, 1\}$, the effective steps do form a RW. Hence, the result of the proposition holds for our RW as well.

The proposition tells us that every time the front is in position $v_2 - 1$, the probability that the message from v_2 will be gathered is $1/v_2$. But every round there will be a message originating from v_2 in $v_2 - 1$ with probability at least $1/2$. So, with probability 1 there will be infinitely many trials with success probability at least $1/v_2$ to gather the event detected by v_2 . We conclude that the probability that the event from v_2 is gathered is 1, for every node v_2 . The following proposition is about a RW with a reflecting barrier. This means that if the message moves from $n - 1$ to n , the next round it moves back to $n - 1$.

Proposition 6.3.4. *Consider a simple symmetric RW with a reflecting barrier in n . Let $\tau(x, n)$ denote the expected time to reach s starting at x . Then $\tau(x, n) = x(2n - x)$. So, the expected time to reach s from position n is n^2 .*

For the RW given by (6.8), the number of steps before a non-zero step is made is geometrically distributed with parameter $1/2$, so it has expectation 2. As a consequence, for this RW we have $\tau(x, n) = 2x(2n - x)$.

If we consider the messages with source node v_2 , then we can view node v_2 as a reflecting barrier. Indeed, consider the situation that all nodes to the left of v_2 are circles. Since v_2 is the source of the square messages, it has a square in its sender cache with probability $1/2$. Hence, with probability $1/2$ it sends a square to node $v_2 - 1$, so that the next round, $v_2 - 1$ is a square with

probability $1/2$. This is exactly the behaviour of the RW (6.8) with a reflecting barrier in v_2 .

Until time t , messages from v_2 move to the sink independent of messages from v_1 . From time t onward, messages from v_2 move towards the sink at a speed comparable to a RW with step size distribution (6.8). Hence, by Proposition 6.3.4, the expected time to reach the base station is roughly $2v_2^2$.

For the case $v_2 - v_1 < t_1 - t_2$, we can use similar arguments to find that $\mathbb{P}(A_1) = \mathbb{P}(A_2) = 1$, and to find the time to gather the event at v_1 . Finally, we should remark that in our analysis we have ignored that the event at node v_1 may form an extra obstacle for messages from v_2 : since v_1 always has a circle in its receiver cache, it is (slightly) more difficult for the squares to pass this node than to pass a normal node.

The 2D grid with cache size one and two events

Given is a 2-dimensional grid of size $\sqrt{n} \times \sqrt{n}$ for some integer \sqrt{n} and with base station s located at one of the corners, see also Figure 6.5. We assume that the size of the sender cache, c , is 1 for each node, and that the broadcast radius r , is 1 as well. We begin by making the observation that in case there is only one fire, the behaviour of the system is equivalent to first-passage percolation, see also Section 6.3. We are interested in the probability that if two events occur, both events are gathered. We consider this probability for the cache strategy **TIMESTAMP DELETION**. The notation is the same as in the previous section, so two events are detected at nodes v_1 and v_2 , respectively, and without loss of generality we assume that v_1 is closer to the sink than v_2 .

Let $\Delta := v_2 - v_1 + t_2 - t_1$. So Δ can be viewed as the time difference between the first arrival at the base station of messages sent from v_1 and v_2 , if both messages are sent independently ($c \geq 2$). From the observations in subsection 6.3 it follows that $\mathbb{P}(A_1) = 1$. For $\mathbb{P}(A_2)$ we consider the case $q = 1$. In this case, the difference in distance to the origin between the two initial points fully determines whether message 1 will reach the origin. This leads to the following theorem:

Theorem 6.3.5. *If $q = 1$, then $\mathbb{P}(A_1) = 1$ and*

$$\mathbb{P}(A_2) = \begin{cases} 1 & \text{if } \Delta < 0 \\ 0 & \text{if } \Delta \geq 3. \end{cases}$$

Proof. As v_1 is closer to the sink than v_2 , a message of v_1 that is sent over a shortest $v_1 - s$ path is always forwarded towards the sink, because for any node u on this path its timestamp is later than that of a message from v_2 at this node u . Hence, $\mathbb{P}(A_1) = 1$. If $\Delta < 0$, then the first message sent from v_2 arrives at each node of a shortest $v_2 - s$ path before a message from v_1 can reach this node. Hence, a message from v_2 arrives at s before a message from v_1 , thus $\mathbb{P}(A_2) = 1$.

Consider a message from v_2 , sent at time t'_2 , that reaches a neighbour of s in round t . Since s is in the corner of the grid, the distance between neighbours

of s is at most 2. Hence, for all $\Delta \geq 3$, there exists a message from v_1 , sent at time $t'_1 > t'_2$, that reaches the same neighbour of s in round t . Since messages from v_1 have a timestamp later than those of v_2 for every neighbour of s , no message sent from v_2 reaches s . If $\Delta = 1$ or $\Delta = 2$, then the probability depends on the position of v_1 relative to v_2 . \square

From this theorem we may derive that using cache strategy **TIMESTAMP DELETION** both events are gathered when the first message of the event which was detected furthest (v_2) could have reached the base station before the first message of v_1 . On the other hand, if the message from v_2 could only have reached the sink at least 3 rounds later, it never reaches the sink. As in the case of the 1D grid, this demonstrates that **TIMESTAMP DELETION** is not a particular good cache strategy if we wish to detect all events.

6.4 Simulations

In this section, we give simulation results for the case with node failures, i.e., we assume $p \leq 1$ throughout this section. In the first paragraph, we consider the problem on a 1D grid when there is a cache size of 1 and there are multiple events. We are interested in the probability that events are gathered under the cache strategy **RANDOM DELETION**. In the second subsection, we consider the same problem on a 2D square grid and a 2D hexagonal grid.

The 1D grid with cache sizes of one and multiple events

Given is a 1D grid of sensor nodes $s, 1, \dots, n$ with base station s ; node i is at distance i from s . We are interested in the influence of q and r on the message completion times for the cache strategy **RANDOM DELETION** when there are multiple events. To this end, we have developed a simulation to analyse these completion times for several arbitrarily generated events.

Given are four events 1, 2, 3, and 4, such that event j is detected at time 0 by node v_j ; the distance of v_j to the sink is $30j$. We assume that nodes do not fail, i.e., $p = 1$.

First consider the case $r = 1$, where each sensor can only broadcast to adjacent nodes. The left picture of Figure 6.9 depicts the outcome of a single simulation run for this case. Each vertical 1D grid in the picture represents the message each sensor in the 1D grid transmits at time t . For instance, until time $t = 20$ each message is broadcast through the 1D grid without any problems; after time $t = 20$, the messages of the second event, coloured black, are blocked by a message of the first event, coloured light gray. When we say a message is blocked we mean that it is not sent further towards the sink. From the figure, we see directly that using the cache strategy **RANDOM DELETION** results in a poor performance of the completion times of the events 2, 3 and 4, whereas messages of event 1 reach the sink without any delay. It seems that event 1 blocks the message of the other events.

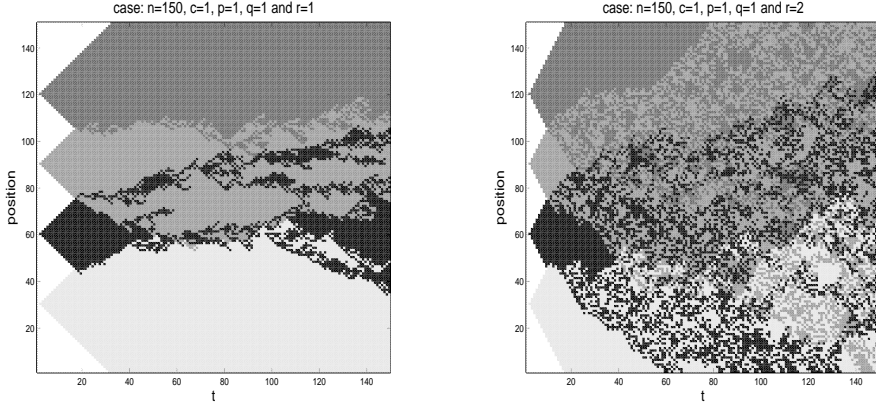


Figure 6.9: Simulation run for the four events starting at $t = 0$ for $r = 1$ (left) and $r = 2$ (right). Each event generates its own unique message, identified by a unique colour. If a sensor did not broadcast any message, then the colour is white. The horizontal axis gives the time (in rounds).

This image changes drastically when we consider a larger broadcast radius, namely $r = 2$. The outcome of a single run is presented in the right picture of Figure 6.9. In this case, the messages become more mixed and as a result also messages from events 2 and 3 reach the sink, within 140 rounds. In particular, we can see that some message of the second event overtakes messages of the first event.

An overview of these observations is plotted in Figure 6.10, which is based on 1000 simulation runs. Note that the horizontal axis in the two figures differs. In the upper figure, representing the situation with $r = 1$, we see that the completion time of event 2 is in general quite large. This gets even worse if we consider the situation of $q = 0.95$, i.e., the case where communication is prone to failures. Although the completion time of event 1 is hardly affected, the completion time of event 2 increases substantially due to the broadcast failures. The lower picture of Figure 6.10, representing the situation with $r = 2$, demonstrates the strongly decreased completion times of event 2. Note also that the impact of broadcast failures (i.e., the $q = 0.95$ case) is smaller than in the $r = 1$ case.

However, in practice it is not always possible to extend the broadcast range to increase the performance. Therefore, another approach would be to change the cache strategy such that messages become more intertwined and in this way keep completion times small. The idea is that a sensor refrains from transmitting the same message all the time and this is formulated in the alternative cache strategy RANDOM DELETION+.

- RANDOM DELETION+: Messages are randomly deleted from the sender cache. The selection of messages to be copied from the receiver cache to the sender

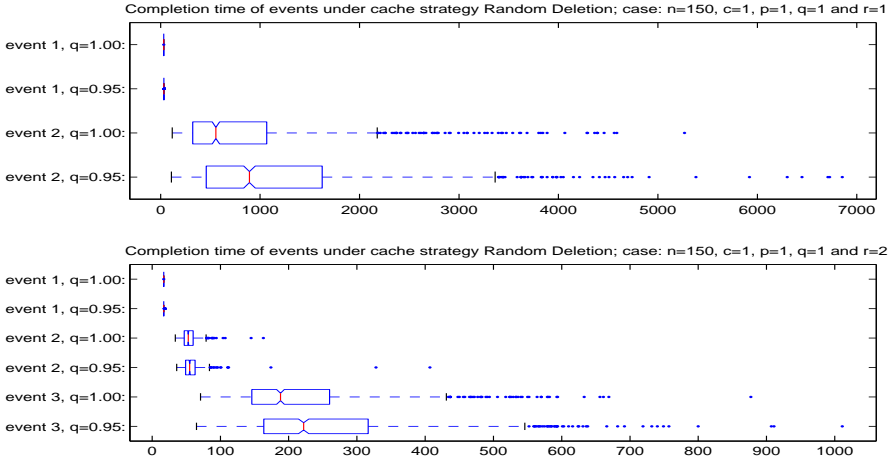


Figure 6.10: The completion time of messages in the form of a box plot. The box has lines at the lower quartile, median, and upper quartile values. The whiskers are lines extending from each end of the box to show the extent of the rest of the data. Outliers are data with values beyond the ends of the whiskers. The whisker extends to the most extreme data value within $1.5 \cdot \text{IQR}$ of the box, where IQR is the width of the interval that contains the middle 50 % of the data. The horizontal axis gives the time (in rounds).

cache is as follows: if the receiver cache contains messages related to an event whose data was broadcast by the same node in the last round, then these messages get low priority: they can only be copied to the sender cache if all other messages are copied as well.

Note that this strategy does not require extra memory. In Figure 6.11, we depict single simulation runs for $r = 1$ (left) and $r = 2$ (right). For both cases we immediately note an improvement compared to the original strategy RANDOM DELETION. The messages of event 1 and 2 no longer block each other and therefore messages of both events travel to the sink without any delay. Unfortunately, it seems that messages of event 4 are still blocked by the messages of the other events.

The results for the completion times over 1000 simulation runs are plotted in Figure 6.12. Note that here the range is much smaller than in Figure 6.11. Also here, the top and bottom ranges differ. The completion times under strategy RANDOM DELETION+ are clearly much smaller than under RANDOM DELETION. The effect of reception failures on the completion times is also negligible.

Thus, using the alternative cache strategy RANDOM DELETION+, we could improve the completion times of the messages. Unfortunately, if there are more than three events then the completion time of all events degenerates, as in the

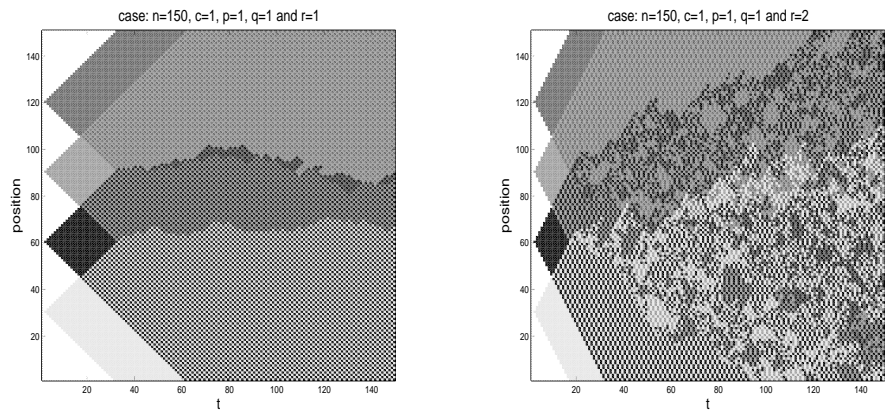


Figure 6.11: Simulations for the cache strategy RANDOM DELETION+. See Figure 6.9 for the interpretation.

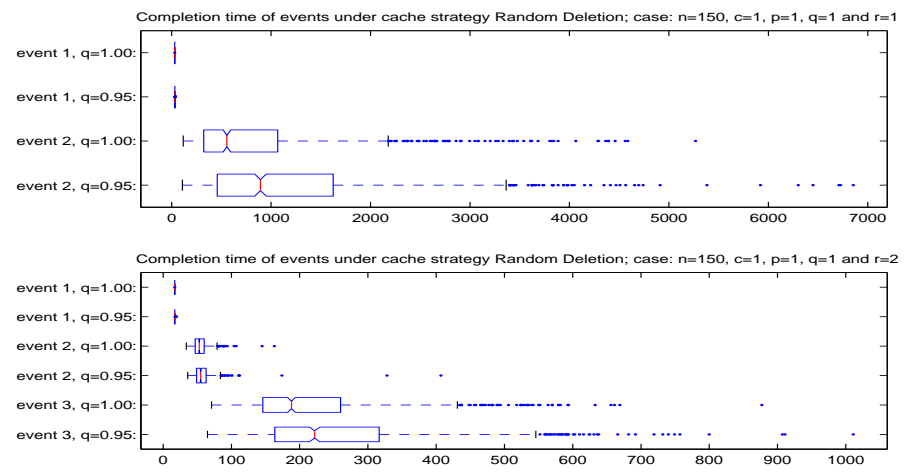


Figure 6.12: See Figure 6.10 for the interpretation, here we use cache strategy RANDOM DELETION+.

case with two events under cache strategy RANDOM DELETION. This could of course be compensated by changing the strategy RANDOM DELETION+ to a more elaborate one, but that would result in a more difficult cache strategy, which might conflict with the aim to keep the strategy as simple as possible.

2D grids with node failures

In the 2D simulations, we consider the area $[0, 100] \times [0, 100]$ covered by sensors with broadcast radius $r = 1$ located at the points of a regular grid. The sensor

placed at $(0,0)$ functions as the base station. We study the data gathering problem for two different grids:

- a square grid with $101 \times 101 = 10201$ sensors numbered $(i, j), 0 \leq i, j \leq 100$.
- a hexagonal grid, where the sensors are located at those points $i(1, 0) + j(0.5, 0.5\sqrt{3})$, with i, j integer, that fall within the area. The total number of sensors is 11658.

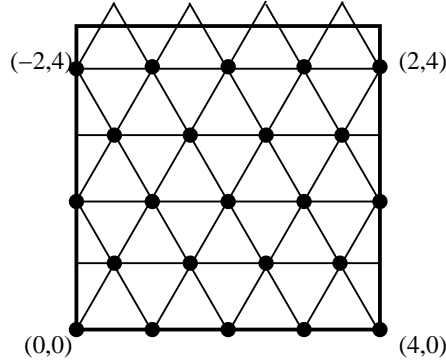


Figure 6.13: Hexagonal grid with 23 sensors covering an area $[0, 4] \times [0, 4]$. The values (i, j) shown are the coordinates of the corners in the hexagonal system. The longest distance to the base station $(0, 0)$ is 6.

The motivation for considering a hexagonal grid, see Figure 6.13, is that each sensor has six neighbours instead of the four in the square grid. We expect this to increase the robustness of the whole system. Furthermore, in the hexagonal grid the distance from the farthest point to the base station is reduced from 200 to 167, thus speeding up the detection of events. A small disadvantage of the hexagonal grid is that more sensors are needed to cover the same area, namely a factor $2/\sqrt{3} \approx 1.16$ more.

In this set of simulations, we study the interference of the messages from the two events. We create two simultaneous events at time $t = 0$, at randomly chosen sensors, and set the cache size $c = 1$. We have run the program 1000 times, each time with a different random number seed. Table 6.2 presents the average number of steps needed to detect the first and second event, for both grids, in case the two events were indeed detected. For $p \leq 0.95$, some runs did not detect any fire. The reason is that failing sensors may cause the graph corresponding to the grid to become disconnected. Furthermore, if an event occurs at a sensor not connected to the base station in the graph, then this event cannot be detected.

The cache strategy we use here is a third variant of RANDOM DELETION, which we call RANDOM DELETION++: the contents of the sender and receive

p	q	Square grid			Hexagonal grid		
		runs	steps 1	steps 2	runs	steps 1	steps 2
1.00	1.00	1000	77.0	141.7	1000	66.1	115.7
	0.95	1000	77.8	146.6	1000	66.2	117.6
0.95	1.00	998	76.2	147.9	996	66.8	117.6
	0.95	998	76.0	148.9	999	67.8	118.5
0.80	1.00	908	80.4	173.1	961	67.0	130.6
	0.95	926	78.4	183.5	940	69.2	135.3

Table 6.2: The number of runs that gathered both events for the strategy RANDOM DELETION++, and the average number of steps needed to gather the first and second event. In total, there were 1000 runs.

cache are merged, duplicates are removed, and messages are randomly deleted until c messages are left. These are then stored in the sender cache. This strategy treats all locally known messages equally (after removal of duplicates), and is not biased towards deleting messages from the sender cache.

The results of Table 6.2 show that the hexagonal grid leads to faster gathering for both events. In particular, the event farthest away from the base station is detected earlier, and its detection time is less affected by failing sensors or failing communications. Note that the ratio of the average gathering times for event 1 corresponds to the ratio of the number of nodes in both networks. A surprising finding is that on the square grid, failing sensors sometimes seem to speed up the detection of the first event, which may be due to less interference from messages for the second event. This is a mixed blessing, as indeed the second event is detected much later.

6.5 Conclusions and recommendations

The main characteristic of a sensor is its simplicity: a sensor has limited processing capabilities, limited power and a limited cache memory. Our objective was to analyse *simple* cache strategies for data gathering in a sensor network. Hence, they should take into account cache constraints, and not use routing information. Our analysis, which consists of an exact analysis based on probability theory and a heuristic analysis through simulation, demonstrates that there exist simple decentralised strategies allowing sensors to gather data efficiently and robust.

We have analysed two strategies: TIMESTAMP DELETION and RANDOM DELETION. In Section 6.3, we have shown that the simplest strategy, TIMESTAMP DELETION, is clearly inferior to RANDOM DELETION, if the number of events is larger than the cache size. Furthermore, simulations in Section 6.4 suggest that the more complicated strategy, RANDOM DELETION+, increases the performance even more. The decision on what level of complexity is allowed

may depend on the application at hand, and should be made by the designers of the system.

A second parameter of interest is the probability that a broadcast fails, q . We have studied how the expected completion time, i.e., the time to gather an event at the base station, depends on q , and the distance from the event to the base station. Different kinds of applications will put different demands on this completion time. For example, forest fires need to be detected immediately, while noise measurements at airports are allowed to come in days later. Hence, per application the system designer should check what values of q are allowed, and what should be done to make sure that q falls within that range.

A third point to consider is the influence of the broadcast range, r . Obviously, the larger the broadcast range, the better. However, due to restrictions on the battery power, only a limited broadcast range may be feasible. The calculations in Section 6.3 reveal the effective step size per round as a function of the broadcast range (and the failure probability q). These results are illustrated and complemented by the simulations in Section 6.4, which show the effect of the broadcast range on the gathering time of events. Again, the demands on the speed by which messages travel through the network should determine how much should be invested in increasing the broadcast range.

Finally, we have considered the influence of the layout of the sensor network in two dimensions. The simulations in Section 6.4 show that a 2D hexagonal layout of sensors is superior to a 2D square layout, both in terms of detection speed and robustness.

We conclude that the performance of a sensor network depends on many parameters. We have tried to describe this performance by analysing some examples of networks. Using this analysis, a system designer could determine the influence of the different parameters. An analysis of the application at hand should reveal which demands both the sensors and the sensor network as a whole have to meet. Combining these two analyses should then yield good and attainable parameter choices.

Acknowledgements We would like to thank Bert Bos (Chess) for providing us with valuable background information on sensor networks, especially concerning technical restrictions and current algorithms. We thank Malwina Luczak for her contribution to the discussions during the week of the Study group.

6.6 Bibliography

- [1] Chess. Presentation, Bert Bos, “Gossiping to optimality”. In *Mathematics with Industry*. TU/e, Eindhoven, The Netherlands, 30-01-2006.
- [2] W. Feller. *An introduction to probability theory and its applications*. Wiley, New York, second edition, 1960.

- [3] P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics*, 53(1-3):79–133, 1994.
- [4] G. Grimmett. *Percolation*. Springer-Verlag, Berlin, second edition, 1999.
- [5] S. Hedetniemi, T. Hedetniemi, and A. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18:319–349, 1988.
- [6] M. Ilyas and I. Mahgoub. *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. CRC Press, Boca Raton, 2004.
- [7] K. Pahlavan and A. Levesque. *Wireless information networks*. Wiley-Interscience, New York, NY, USA, 1995.
- [8] R. T. Smythe and J. C. Wierman. *First-passage percolation on the square lattice*, volume 671 of *Lecture Notes in Mathematics*. Springer, Berlin, 1978.
- [9] W. Su, E. Cayirci, and O. Akan. Overview of communication protocols for sensor networks. In *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. CRC Press, Boca Raton, 2004.