

Modeling pollen-tube growth

Johannes Horlemann

09.05.2005

Seminar report for Optimization of Biotechnical Processes
Ruprecht-Karls-University of Heidelberg
Institute for Scientific Computing (IWR)
Supervisors: Moritz Diehl, Dirk Hartmann

Contents

Abstract	3
1 Introduction	3
2 Biological Background of the Model	3
2.1 Rac GTPase as central regulating protein	3
2.2 Effectors of Rac-GTPase	4
3 Modeling	5
3.1 Growth Speed	5
3.2 Diffusion and Transport	6
3.3 Assumptions	7
3.4 The Complete Model	8
3.5 Travelling waves	8
3.6 Implementation	10
3.6.1 Full Model	10
3.6.2 Creating a simplified Model	10
4 Results	10
4.1 Full Model	10
4.2 Simplified Model	10
5 Conclusion and Outlook	20
6 Appendix	23
6.1 Program Code of the complete Model implemented in MUSCOD-II	23
6.2 Program Code of the simplified Model implemented in MUSCOD-II	34
6.3 Program Code of the simplified Model implemented in Matlab 6.5	42
Acknowledgments	45
References	45

Abstract

Polarized cell growth is a very fascinating process and concerns the growth of root hairs and pollen-tubes. By now, only a few of the complex biological connections have been investigated, but a monomeric GTPase called Rac seems to be the central switch in regulating pollen-tube growth. On the basis of known pathways, a mathematical model was introduced which describes processes that lead to the polarized cell growth which is observed. The model was then implemented in MUSCOD-II [5] for simulation purposes. Some difficulties in finding a valid solution could be observed, because the parameters could not be chosen systematically. Therefore, the model has been further simplified to give evidence about the correctness of the model, which could be shown. The simulation results of the simplified model show the expected tendencies which base on the experimental biological data.

1 Introduction

Pollen-tube growth as one example of polarized cell growth plays a critical role for plant reproduction and is crucial to further development and survival of the whole organism [2]. Pollen-tubes are tube-like, highly elongated cells with a large surface. They grow into a specific direction to deliver male sperms to the ovule for fertilization. For this purpose cells of this type only grow at the tip of the tube and can reach an incredible growth speed of up to 200nm/s [2].

The process of fertilization starts with a pollen grain which lands on the stigma surface of the plant. In direct neighborhood to a germination aperture a pollen-tube grows rapidly through the surface and finally reaches the ovule for fertilization. As mentioned above, the growth of pollen-tubes is restricted to the apex of the tube where golgi-vesicles fuse with the plasma membrane by localized exocytosis. This leads to a polar assembly of new plasma membrane and cell wall regions at the tip of the pollen-tube [1]. This tip growth is internally controlled by a set of ion gradients (e.g. a tip-focused Ca^{2+} gradient), signalling proteins, protein kinases, and cytoskeleton dynamics [1, 2].

2 Biological Background of the Model

Describing pollen-tube growth with a mathematical model, we based our focus on following connections:

2.1 Rac GTPase as central regulating protein

The Rac GTPase belongs to the RHO family of monomeric GTPases and is a guanine nucleotide-binding protein [1]. It is bound to the plasma membrane through a C-terminal prenyl tail. As shown in Figure 1 the inactive state of Rac is bound to GDP.

It becomes activated by a guanine-nucleotide exchange factor (GEF) which exchanges the GDP for a GTP. In its activated state, Rac can bind to different effectors and activate them. As mentioned above, Rac has a GTPase activity and can spontaneously decompose GTP to GDP and an inorganic Phosphate (P_i). This effect is supported by a GTPase activating protein (GAP). Additionally, the Rac-GDP can bind to a guanine-nucleotide dissociation inhibitor (GDI) and forms a complex which is not bound to the plasma membrane (PM). Experimental data [3] suggests the localization of Rac, GEF, GAP and GDI as shown in Figure 2, where activated Rac is only localized at the tip of the tube.

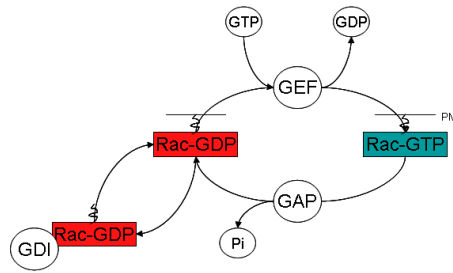


Figure 1: Regulatory cycle of Rac.

2.2 Effectors of Rac-GTPase

A definite class of effectors of the Rac-GTPase are the phosphatidylinositol monophosphate kinases which generate Phosphatidylinositol-4,5-bisphosphate ($PI(4,5)P_2$) anchored to the plasma membrane [1]. This leads to the assumption that $PI(4,5)P_2$ is only available in measurable concentrations in regions with activated Rac (Rac-GTP), e.g. at the tip of the tube. Hence, we can neglect $PI(4,5)P_2$ at regions without Rac-GTP, which is shown in Figure 2.

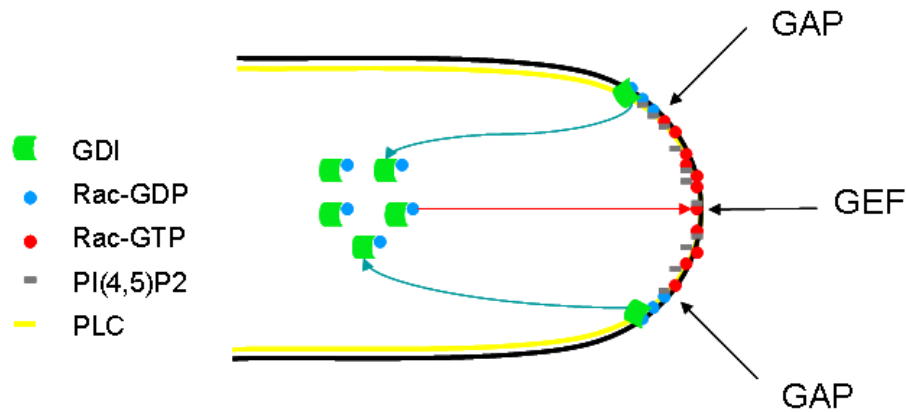


Figure 2: Localization of GAP, GEF, GDI, PLC, and $PI(4,5)P_2$.

Another effector of Rac-GTPase is phospholipase C (PLC) which is possibly regulating the establishment of a tip-focused Ca^{2+} gradient. For this purpose, PLC hydrolyzes $PI(4,5)P_2$ to inositol-1,4,5-triphosphate (IP_3) and diacylglycerole (DAG). Then, IP_3 opens Ca^{2+} channels at the endoplasmic reticulum (ER) which leads to an Ca^{2+} influx into the cell plasma (see Figure 3). DAG is still bound to the plasma membrane and can now bind GAP. Thereby, GAP is not located at the tip of the tube, but at its borders. Experimental data [3] shows, that PLC only occurs beneath the apex area of the tube, which is also shown in Figure 2.

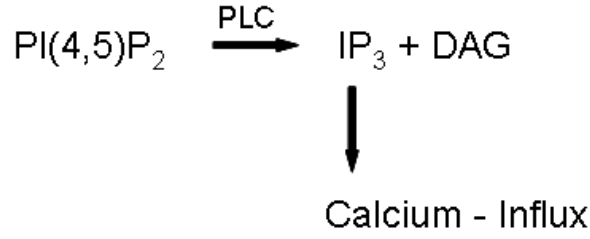


Figure 3: Hydrolyzation of $PI(4,5)P_2$ by PLC.

3 Modeling

3.1 Growth Speed

The cytoplasm and the membrane are modeled by one dimensional domains, \mathcal{C} and \mathfrak{M} respectively. For both we consider a coordinate system which is moving with the tip speed V_{tip} , i.e. the origin is always at the tip.

By inclusion of new membrane material at the tip, the membrane length increases. As the coordinate system is fixed at the tip this implies an effective transport mechanism in the membrane as shown in Figure 4. Hence it follows that the growth speed at the tip (i.e. at $x=0$) is zero and rises with increasing x .

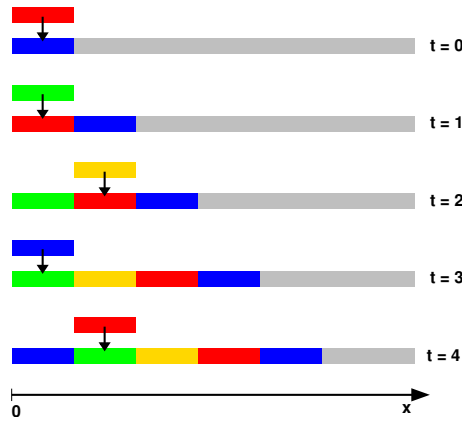


Figure 4: Schematic illustration of the membrane growth.

Due to the incompressibility of the membrane, the following conservation law must hold for any interval $[0, l] \subset \mathfrak{M}$

$$V(l, t) = \int_{[0, l]} (p_1^*(c_{\text{Rac-GTP}}^m, c_{\text{PI4,5-P}_2}^m, c_{\text{IP}_3}^c) c_{\text{vesicles}}^c(x, t)) dx, \quad (3.1)$$

or equivalently

$$0 = -\partial_x V + p_1^*(c_{\text{Rac-GTP}}^m, c_{\text{PI4,5-P}_2}^m, c_{\text{IP}_3}^c) c_{\text{vesicles}}^c \quad (3.2)$$

with the boundary condition $V(0, t) = 0$ must hold for all $x \in \mathfrak{M}$. Annotation: c_{\dots}^m describe membrane concentrations and c_{\dots}^c describe cytoplasmic concentrations.

We neglected the fact that some membrane material is removed from the membrane, decreasing the speed, e.g. we neglect p_2 (see Figure 5). All membrane bound factors will

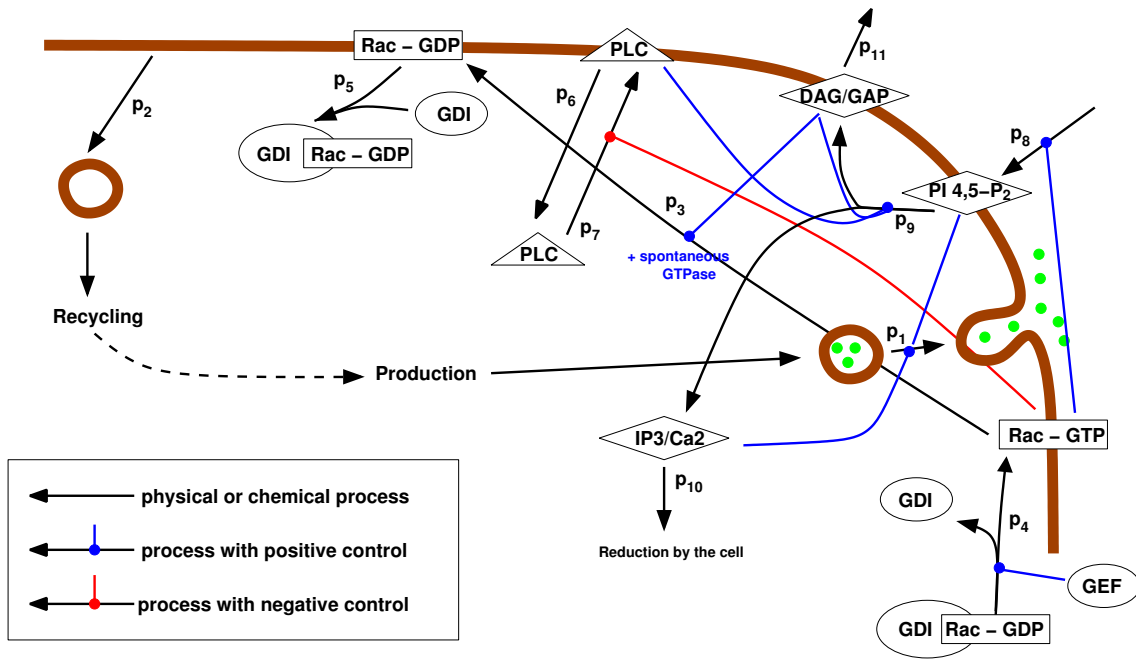


Figure 5: Illustration of the model.

be transported with the speed of the membrane material $V(x, t)$. For simplicity we will only consider a constant vesicle concentration at the tip, i.e. we regard $c_{\text{vesicles}}^c(x, t)$ as a constant. The constant can be absorbed into the kinetics, i.e.

$$p_1 = c_{\text{vesicles}}^c p_1^* (c_{\text{Rac-GTP}}^m, c_{\text{PI4,5-P}_2}^m, c_{\text{IP}_3}^c). \quad (3.3)$$

$$\iff \partial_x V = +p_1 (c_{\text{Rac-GTP}}^m, c_{\text{PI4,5-P}_2}^m, c_{\text{IP}_3}^c) \quad (3.4)$$

For p_1 we assume the following linear relation:

$$p_1 = k_0 c_{\text{Rac-GTP}}^m c_{\text{PI4,5-P}_2}^m c_{\text{IP}_3}^c. \quad (3.5)$$

3.2 Diffusion and Transport

Diffusion describes processes where only by random molecular motion matter is transported from one part into another part of a system [4]. Experimentally, diffusion can be observed as follows: pouring clear water in a cylindrical vessel half filled with a iodine solution so that no convection currents occur, one can watch the upper part of the solution becoming slightly colored while the color of the lower part fades. After a while, the whole solution is homogenously colored. Thus leads to the assumption that a transfer of iodine molecules from the lower to the upper part of the vessel took place. But every single molecule undergoes molecular motion which is completely random and independent of other molecules in the neighborhood. As a result of collision with molecules of the solvent the molecules do not have a preferred direction of motion into regions of higher or lower concentration. Provided that every molecule is in random motion, a connection must be found to explain the directed transfer of iodine into regions of lower concentration. If one chooses a thin horizontal section of the solution and a same-sized section above and

below of the section, on average the same definite fraction of molecules walk from the lower section upwards and from the upper section downwards. Due to the fact that the lower section contains a higher concentration of iodine than the upper section, there is a clear transfer from the lower to the upper section.

Following the derivation in [4], the rate of transfer through unit area of a section can be assumed to be proportional to the concentration gradient normal to the section for an isotropic medium, i.e.

$$F_x = -D\partial_x c, \quad (3.6)$$

where F_x is the rate of transfer per unit area of a section, c is the concentration of the considered substance, and D is the diffusion coefficient. Due to the fact that diffusion decreases in direction of increasing concentration a negative sign can be found in equation (3.6).

Including a transport term, equation (3.6) changes to

$$F_x = -D\partial_x c + Vc. \quad (3.7)$$

The time dependence of the concentration is given by:

$$\partial_t c = D\partial_{xx} c - \partial_x Vc. \quad (3.8)$$

Adapting this equation to our model, we extend equation (3.8) by adding a kinetic term

$$\partial_t c = D\partial_{xx} c - \partial_x Vc + p(c) \quad (3.9)$$

where $p(c)$ is a function describing the production and reduction of the substance. This leads to the basic equation for the model of pollen-tube growth:

$$\partial_t c = D\partial_{xx} c - V\partial_x c - c\partial_x V + p(c). \quad (3.10)$$

3.3 Assumptions

The following assumptions were made to simplify the complexity of biochemical kinetics used for the model:

- As mentioned above, phosphatidylinositol monophosphate kinases are effectors of Rac-GTP and are only activated in regions of existent Rac-GTP. Therefore, we neglect the kinases in our model and use Rac-GTP directly for the interrelations concerning the kinases. Additionally, $PI(4,5)P_2$ is neglected at regions without Rac-GTP (see Section 2.2).
- Because DAG and GAP occur co-located we combined them to DAG/GAP.
- For the Ca^{2+} -release from the ER induced by IP_3 we considered a linear connection and therefore we also combined both as IP_3 .
- For the first stage, we decided to set the vesicle concentration constant, i.e.

$$c_{\text{vesicles}} = \text{const.} \quad (3.11)$$

- We considered the substrate for the production of $PI(4,5)P_2$ to be in a surplus which we neglected it in our model.
- GEF is treated as a given distribution and was included in p_4 .

3.4 The Complete Model

The chemical control mechanisms of the membrane organization are shown in Figure 5 and are reflected by the following mathematical model:

$$\begin{aligned}
\partial_t c_{\text{Rac-GTP}}^m &= -V \partial_x c_{\text{Rac-GTP}}^m - c_{\text{Rac-GTP}}^m \partial_x V - p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m + p_4 c_{\text{GDI-Rac-GDP}}^c \\
\partial_t c_{\text{Rac-GDP}}^m &= -V \partial_x c_{\text{Rac-GDP}}^m - c_{\text{Rac-GDP}}^m \partial_x V + p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
\partial_t c_{\text{GDI-Rac-GDP}}^c &= D_1 \partial_{xx} c_{\text{GDI-Rac-GDP}}^c - p_4 c_{\text{GDI-Rac-GDP}}^c + p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
\partial_t c_{\text{GDI}}^c &= D_2 \partial_{xx} c_{\text{GDI}}^c + p_4 c_{\text{GDI-Rac-GDP}}^c - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
\partial_t c_{\text{PLC}}^m &= -V \partial_x c_{\text{PLC}}^m - c_{\text{PLC}}^m \partial_x V - p_6 c_{\text{PLC}}^m + p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c \\
\partial_t c_{\text{PLC}}^c &= D_3 \partial_{xx} c_{\text{PLC}}^c + p_6 c_{\text{PLC}}^m - p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c \\
\partial_t c_{\text{PI}4,5\text{-P}_2}^m &= -V \partial_x c_{\text{PI}4,5\text{-P}_2}^m - c_{\text{PI}4,5\text{-P}_2}^m \partial_x V + p_8 (c_{\text{Rac-GTP}}^m) - p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m \\
\partial_t c_{\text{IP}_3}^c &= D_4 \partial_{xx} c_{\text{IP}_3}^c + p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m - p_{10} c_{\text{IP}_3}^c \\
\partial_t c_{\text{DAG/GAP}}^m &= -V \partial_x c_{\text{DAG/GAP}}^m - c_{\text{DAG/GAP}}^m \partial_x V + p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m - p_{11} c_{\text{DAG/GAP}}^m.
\end{aligned} \tag{3.12}$$

Therefore, pollen-tube growth is described by the system of partial differential equations 3.2 and 3.12 plus appropriate initial and boundary conditions (see below).

3.5 Travelling waves

Pollen-tube growth has quite stable dynamics, i.e. it can be regarded as travelling waves. These are solutions which appear constant in an appropriately moving coordinate system. Using the coordinate system explained above, we can therefore set the time derivatives in our model to zero. This yields the following system of first and second order ordinary differential equations:

$$\begin{aligned}
0 &= -V \partial_x c_{\text{Rac-GTP}}^m - c_{\text{Rac-GTP}}^m \partial_x V - p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m + p_4 c_{\text{GDI-Rac-GDP}}^c \\
0 &= -V \partial_x c_{\text{Rac-GDP}}^m - c_{\text{Rac-GDP}}^m \partial_x V + p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
0 &= D_1 \partial_{xx} c_{\text{GDI-Rac-GDP}}^c - p_4 c_{\text{GDI-Rac-GDP}}^c + p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
0 &= D_2 \partial_{xx} c_{\text{GDI}}^c + p_4 c_{\text{GDI-Rac-GDP}}^c - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m \\
0 &= -V \partial_x c_{\text{PLC}}^m - c_{\text{PLC}}^m \partial_x V - p_6 c_{\text{PLC}}^m + p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c \\
0 &= D_3 \partial_{xx} c_{\text{PLC}}^c + p_6 c_{\text{PLC}}^m - p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c \\
0 &= -V \partial_x c_{\text{PI}4,5\text{-P}_2}^m - c_{\text{PI}4,5\text{-P}_2}^m \partial_x V + p_8 (c_{\text{Rac-GTP}}^m) - p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m \\
0 &= D_4 \partial_{xx} c_{\text{IP}_3}^c + p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m - p_{10} c_{\text{IP}_3}^c \\
0 &= -V \partial_x c_{\text{DAG/GAP}}^m - c_{\text{DAG/GAP}}^m \partial_x V + p_9 (c_{\text{PLC}}^m) c_{\text{PI}4,5\text{-P}_2}^m - p_{11} c_{\text{DAG/GAP}}^m.
\end{aligned} \tag{3.13}$$

This leads to the following system of equations:

$$\begin{aligned}
\partial_x c_{\text{Rac-GTP}}^m &= \frac{(-c_{\text{Rac-GTP}}^m \partial_x V - p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m + p_4 c_{\text{GDI-Rac-GDP}}^c)}{V} \\
\partial_x c_{\text{Rac-GDP}}^m &= \frac{(-c_{\text{Rac-GDP}}^m \partial_x V + p_3 (c_{\text{DAG/GAP}}^m) c_{\text{Rac-GTP}}^m - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m)}{V} \\
\partial_{xx} c_{\text{GDI-Rac-GDP}}^c &= \frac{(p_4 c_{\text{GDI-Rac-GDP}}^c - p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m)}{D_1} \\
\partial_{xx} c_{\text{GDI}}^c &= \frac{(-p_4 c_{\text{GDI-Rac-GDP}}^c + p_5 c_{\text{GDI}}^c c_{\text{Rac-GDP}}^m)}{D_2} \\
\partial_x c_{\text{PLC}}^m &= \frac{(-c_{\text{PLC}}^m \partial_x V - p_6 c_{\text{PLC}}^m + p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c)}{V} \\
\partial_{xx} c_{\text{PLC}}^c &= \frac{(-p_6 c_{\text{PLC}}^m + p_7 (c_{\text{Rac-GTP}}^m) c_{\text{PLC}}^c)}{D_3} \\
\partial_x c_{\text{PI4,5-P}_2}^m &= \frac{(-c_{\text{PI4,5-P}_2}^m \partial_x V + p_8 (c_{\text{Rac-GTP}}^m) - p_9 (c_{\text{PLC}}^m) c_{\text{PI4,5-P}_2}^m)}{V} \\
\partial_{xx} c_{\text{IP}_3}^c &= \frac{(-p_9 (c_{\text{PLC}}^m) c_{\text{PI4,5-P}_2}^m + p_{10} c_{\text{IP}_3}^c)}{D_4} \\
\partial_x c_{\text{DAG/GAP}}^m &= \frac{(-c_{\text{DAG/GAP}}^m \partial_x V + p_9 (c_{\text{PLC}}^m) c_{\text{PI4,5-P}_2}^m - p_{11} c_{\text{DAG/GAP}}^m)}{V}.
\end{aligned} \tag{3.14}$$

For the functions p_3 to p_{11} we consider:

$$\begin{aligned}
p_3 (c_{\text{DAG/GAP}}^m) &= k_1 + k_2 \frac{c_{\text{DAG/GAP}}^m}{k_3 + k_4 c_{\text{DAG/GAP}}^m} \\
p_4 &= k_5 \\
p_5 &= k_6 \\
p_6 &= k_7 \\
p_7 (c_{\text{Rac-GTP}}^m) &= k_8 \exp(-k_9 c_{\text{Rac-GTP}}^m) \\
p_8 (c_{\text{Rac-GTP}}^m) &= k_{10} \frac{c_{\text{Rac-GTP}}^m}{k_{11} + k_{12} c_{\text{Rac-GTP}}^m} \\
p_9 (c_{\text{PLC}}^m) &= k_{13} \frac{c_{\text{PLC}}^m}{k_{14} + k_{15} c_{\text{PLC}}^m} \\
p_{10} &= k_{16} \\
p_{11} &= k_{17},
\end{aligned} \tag{3.15}$$

where $k_1 \dots k_{17} = \text{const.}, \in \mathbb{R}$.

For each of the first order equations we need to specify one boundary condition. Due to symmetry reasons we require for all concentrations

$$\partial_x c_{\dots}^m/c|_{x=0} = 0 \tag{3.16}$$

and for the speed

$$V|_{x=0} = 0. \tag{3.17}$$

The system of ordinary differential equations (3.13), where the values of the constants still have to be defined, together with the boundary conditions (3.16) and (3.17) describe the complete dynamics of the traveling wave solutions.

3.6 Implementation

3.6.1 Full Model

For the simulation of the model we used MUSCOD-II [5]; for the program code see Section 6.1. States, parameters and initial conditions used are shown in Table 1 and 2. All values of the parameters were not chosen systematically due to insufficient knowledge about their real values. Additionally, a simulation window from 0 to 1 was used.

For the simulation results see Section 4.1. The model turned out to be quite unstable and we had difficulties in finding the right parameters, such that the solutions reflect the experimental data. Because of this fact, we decided to simplify the model by adding some more assumptions in order to create a stable system.

3.6.2 Creating a simplified Model

First, we set all cytoplasmic changes to zero, under the assumption that diffusion and transport processes are faster in the cytoplasm than in the membrane. Second, we neglected the boundary conditions (3.16) and defined initial values instead. Third, we forced the system to allow a inclusion of Rac-GTP only for $x < 0.1$, i.e. $k_5 = 0.3$ for $x \leq 0.1$ and $k_5 = 0$ for $x > 0.1$. This is due to the fact that Rac-GTP is only installed at the tip of the pollen tube.

The simplified model was implemented in MUSCOD-II and Matlab 6.5 [6] for the simulation. The altered program code for MUSCOD-II can be seen in Section 6.2, and for Matlab 6.5 in Section 6.3. The states, parameters and initial values used are shown in Tables 3 and 4.

4 Results

4.1 Full Model

As mentioned above, the model turned out to be very unstable. Due to insufficient knowledge of the biological parameters it was difficult to find a solution which was unequal to zero. This solution (see Figures 6 to 19) shows a low explanatory power, so there is no need in discussing every diagram, but a tendency becomes obvious: except for $c_{IP_3}^c$ all concentrations are positive. For $c_{IP_3}^c \geq 0$ the solver (DAESOL) was not able to find a valid solution for the equation system, which is not biologically meaningful.

4.2 Simplified Model

The simulations in MUSCOD-II and Matlab 6.5 show equal results and are shown in Figures 20 to 25. To create these graphs, data points of the simulation were extracted from results-files of MUSCOD-II and were plotted by Microsoft Excel. At position $x=0.1$ a crease is recognizable which is due to the fact that at this position k_5 becomes suddenly zero and that these curves are linear interpolated to the data points. For the interval from 0 to 0.1, 10 data points were prepared and for the interval from 0.1 to 1 18 data points were prepared for the curve fitting. Altogether, the results are very satisfactory and show the tendencies we expected (see Section 2 and Figure 2):

- The curve of $c_{DAG/GAP}^m$ shows a maximum of concentration at $x=0.9$. This underlines the experimental results that it is preferentially located at a defined position (see Figure 2).

Table 1: States of the full model.

State	Designation in MUSCOD-II	State number	Initial values
$c_{\text{Rac-GTP}}^m$	cmRacGTP	xd[0]	1.0475440106315601E-001
$c_{\text{Rac-GDP}}^m$	cmRacGDP	xd[1]	8.5864471844518908E-001
$c_{\text{GDI-Rac-GDP}}^c$	ccGDIRacGDP	xd[2]	1.8356189176726373E-001
$\partial_t c_{\text{GDI-Rac-GDP}}^c$	ccGDIRacGDPdot	xd[3]	7.3217698861876916E-312
c_{GDI}^c	ccGDI	xd[4]	2.9646112281949422E-001
$\partial_t c_{\text{GDI}}^c$	ccGDI dot	xd[5]	7.3217698861876916E-312
c_{PLC}^m	cmPLC	xd[6]	5.6538031758336098E-002
c_{PLC}^c	ccPLC	xd[7]	7.5906614486345794E-002
$\partial_t c_{\text{PLC}}^c$	ccPLCdot	xd[8]	-7.3211906720283413E-312
$c_{\text{PI } 4,5\text{-P}_2}^m$	cmPIP2	xd[9]	1.9176563413111862E+000
c_{IP3}^c	ccIP3	xd[10]	4.2965973428051675E-002
$\partial_t c_{\text{IP3}}^c$	ccIP3dot	xd[11]	7.2385865927268383E-312
$c_{\text{DAG/GAP}}^m$	cmDAGGAP	xd[12]	2.1376596019775718E-001
V	V	xd[13]	1.0000000000000000E-004

Table 2: Parameters of the full model.

Parameter	Designation in MUSCOD-II	Parameter number	Value
k_0	k0	p[0]	0.5
k_1	k1	p[1]	0.5
k_2	k2	p[2]	0.1
k_3	k3	p[3]	1
k_4	k4	p[4]	0
k_5	k5	p[5]	0.3
k_6	k6	p[6]	0.2
k_7	k7	p[7]	0.5
k_8	k8	p[8]	0.4
k_9	k9	p[9]	0.6
k_{10}	k10	p[10]	0.7
k_{11}	k11	p[11]	1
k_{12}	k12	p[12]	0
k_{13}	k13	p[13]	0.6
k_{14}	k14	p[14]	1
k_{15}	k15	p[15]	0
k_{16}	k16	p[16]	0.2
k_{17}	k17	p[17]	0.3
D_1	D1	p[18]	0.5
D_2	D2	p[19]	0.5
D_3	D3	p[20]	0.5
D_4	D4	p[21]	0.5

Table 3: States of the simplified model.

State	Designation in MUSCOD-II	State number	Initial values
$c_{\text{Rac-GTP}}^m$	cmRacGTP	xd[0]	1
$c_{\text{Rac-GDP}}^m$	cmRacGDP	xd[1]	0.1
c_{PLC}^m	cmPLC	xd[2]	0.2
$c_{\text{PI 4,5-P}_2}^m$	cmPIP2	xd[3]	0.9
$c_{\text{DAG/GAP}}^m$	cmDAGGAP	xd[4]	0.1
V	V	xd[5]	0.01

Table 4: Parameters of the simplified model.

Parameter	Designation in MUSCOD-II	Parameter number	Value
k_0	k0	p[0]	0.5
k_1	k1	p[1]	0.5
k_2	k2	p[2]	0.1
k_3	k3	p[3]	1
k_4	k4	p[4]	0
k_5	k5	p[5]	0.3
k_6	k6	p[6]	0.2
k_7	k7	p[7]	0.5
k_8	k8	p[8]	0.4
k_9	k9	p[9]	0.6
k_{10}	k10	p[10]	0.7
k_{11}	k11	p[11]	1
k_{12}	k12	p[12]	0
k_{13}	k13	p[13]	0.6
k_{14}	k14	p[14]	1
k_{15}	k15	p[15]	0
k_{16}	k16	p[16]	0.2
$c_{\text{GDI-Rac-GDP}}^c$	ccGDIRacGDP	p[17]	1
c_{GDI}^c	ccGDI	p[18]	1
c_{PLC}^c	ccPLC	p[19]	1
c_{IP3}^c	ccIP3	p[20]	1

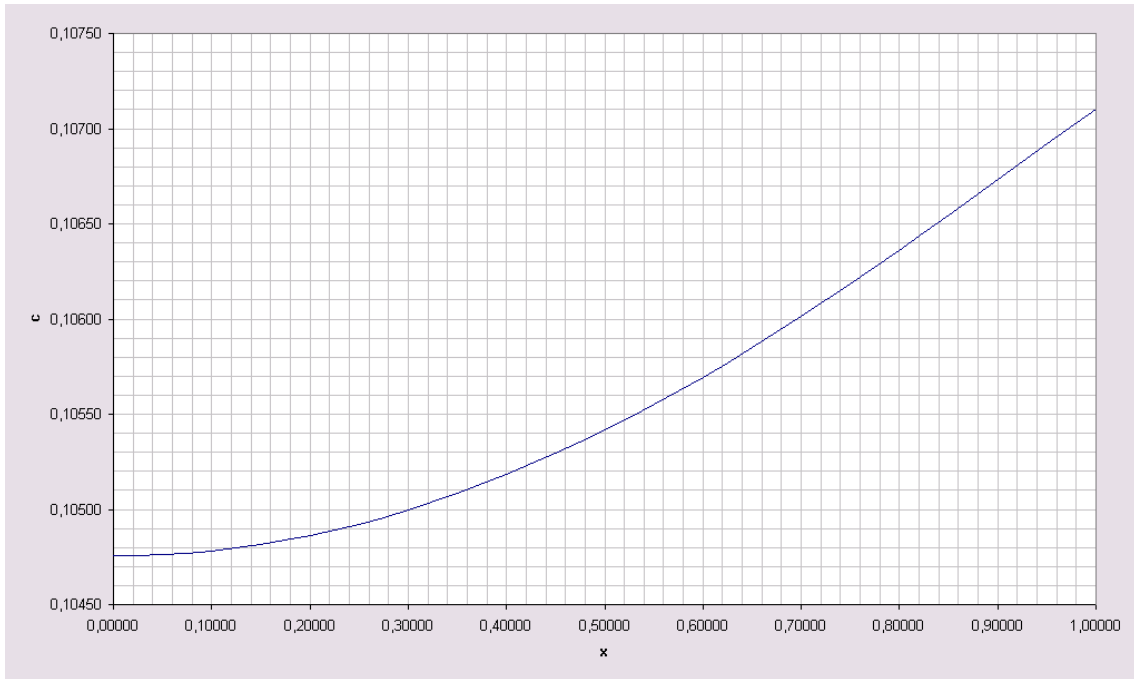


Figure 6: Full model: Concentration profile of $c_{\text{Rac-GTP}}^m$.

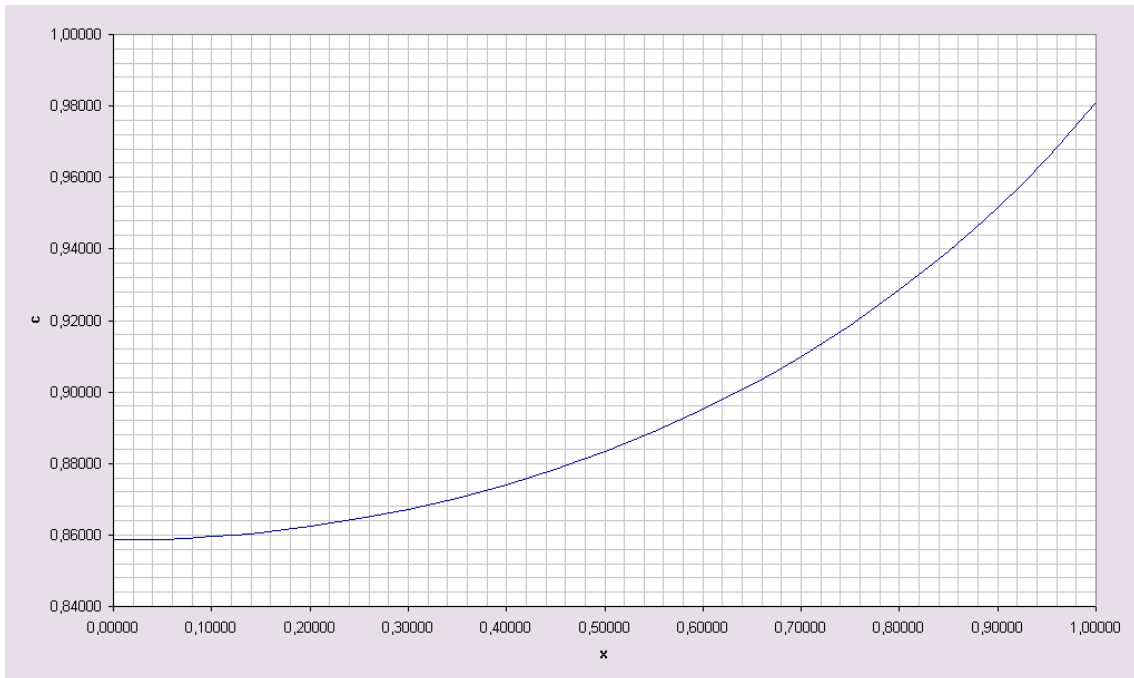


Figure 7: Full model: Concentration profile of $c_{\text{Rac-GDP}}^m$.

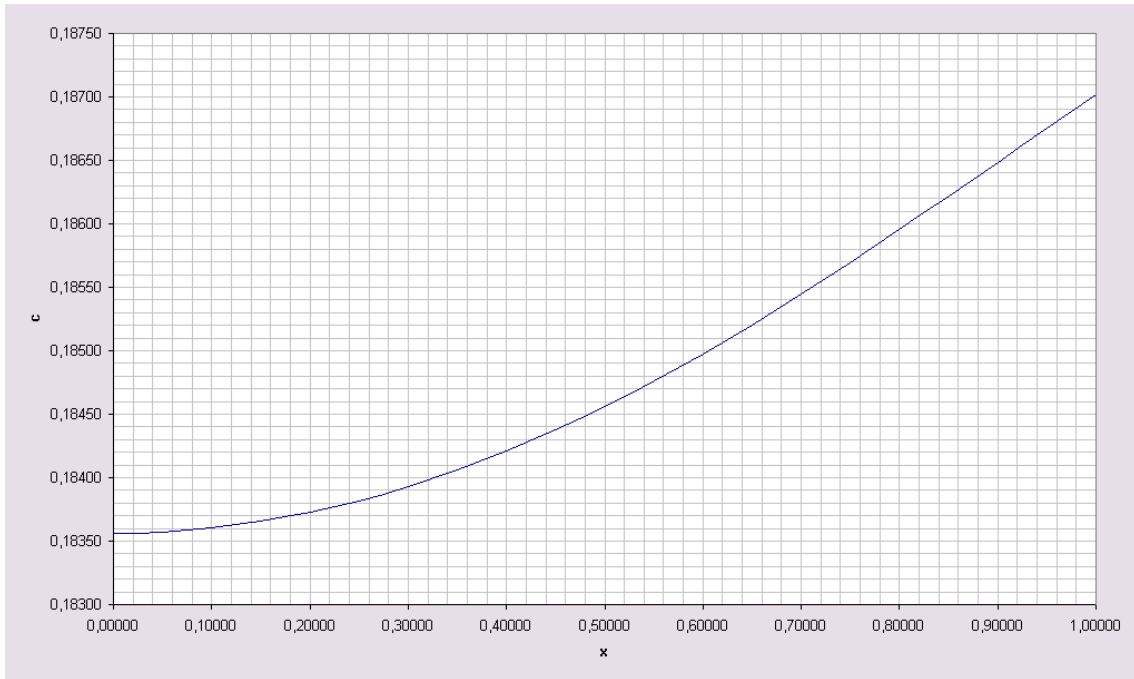


Figure 8: Full model: Concentration profile of $c_{\text{GDI-Rac-GDP}}^c$.

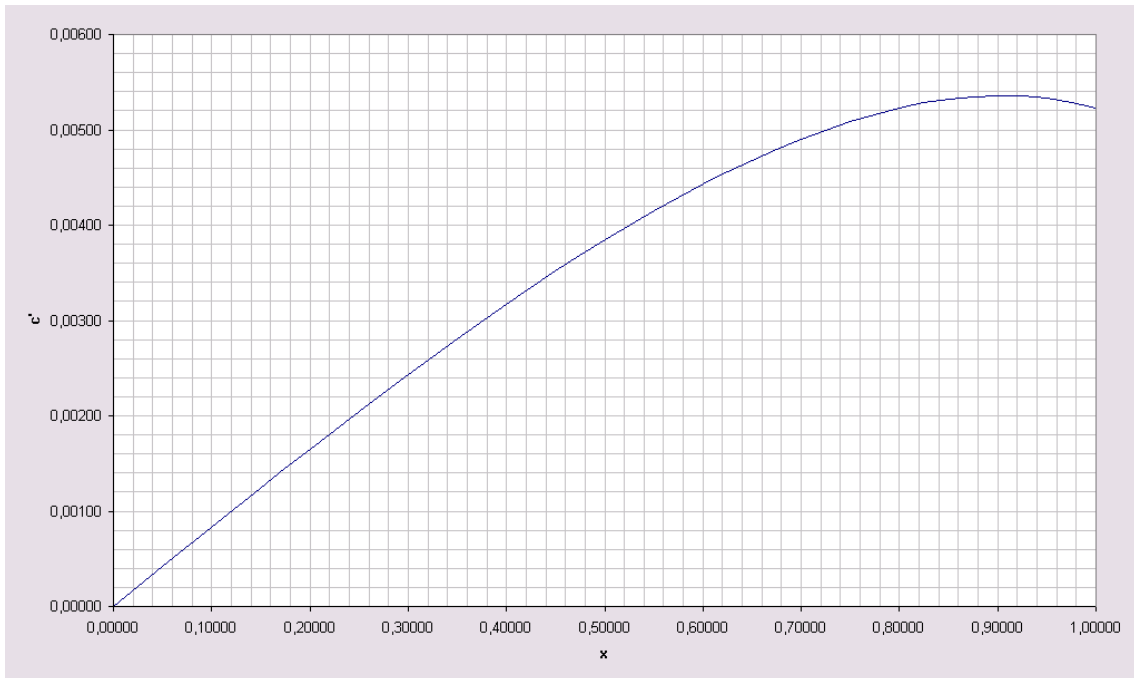


Figure 9: Full model: Profile of $\partial_x c_{\text{GDI-Rac-GDP}}^c$.

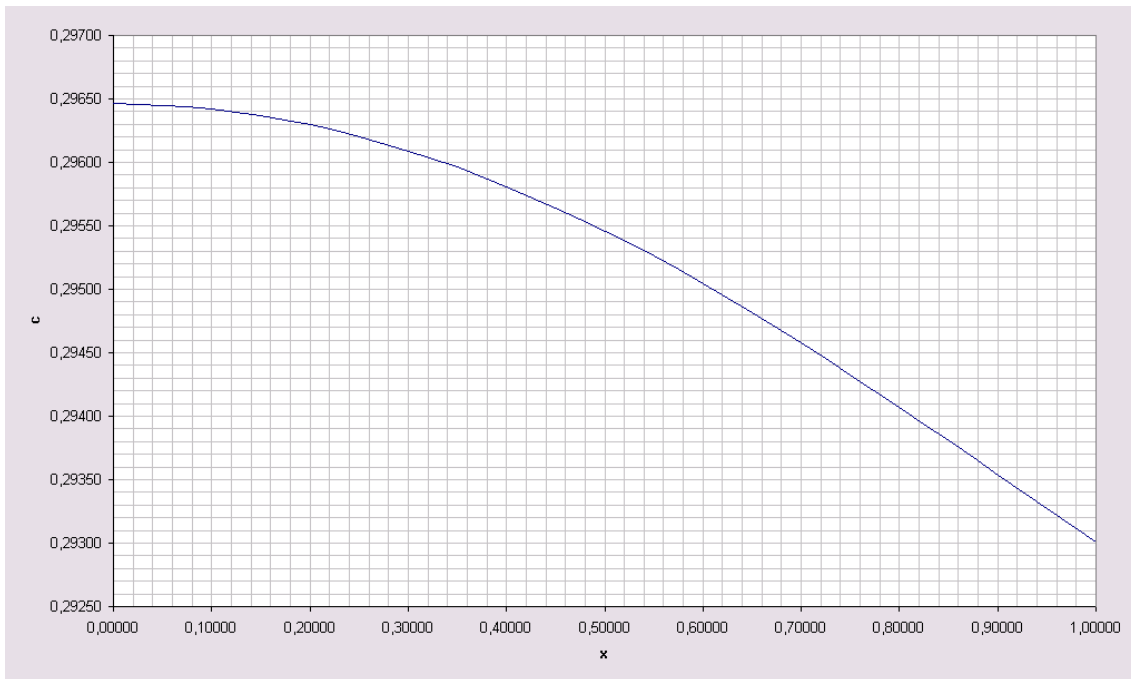


Figure 10: Full model: Concentration profile of c_{GDI}^c .



Figure 11: Full model: Profile of $\partial_x c_{GDI}^c$.

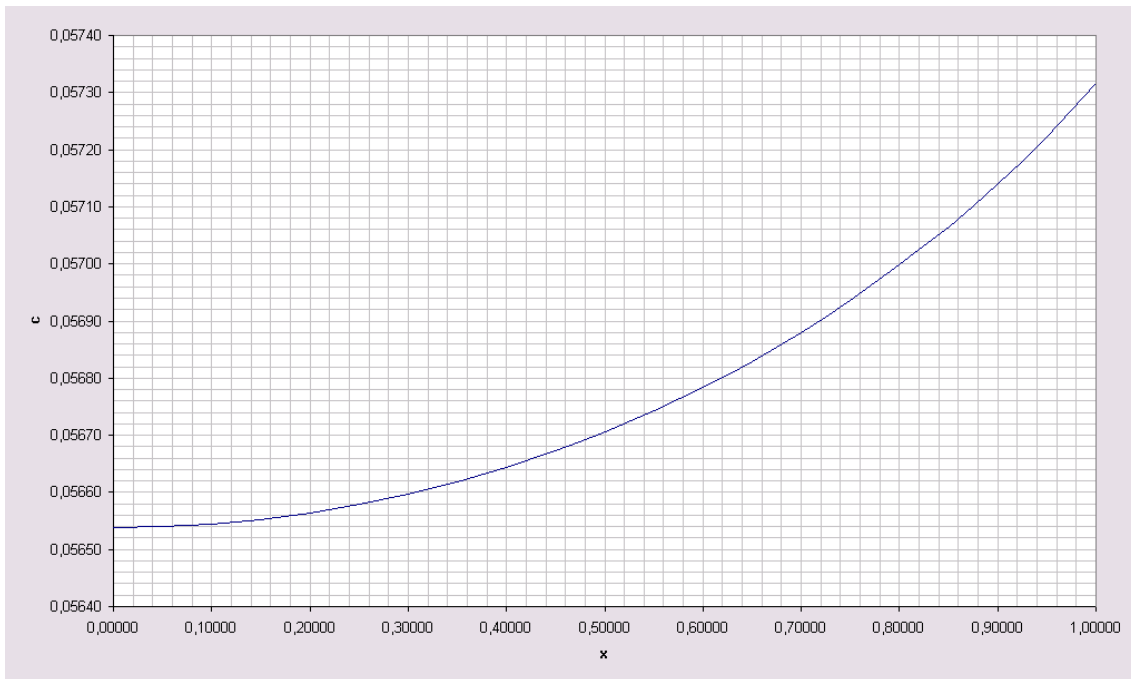


Figure 12: Full model: Concentration profile of c_{PLC}^m .

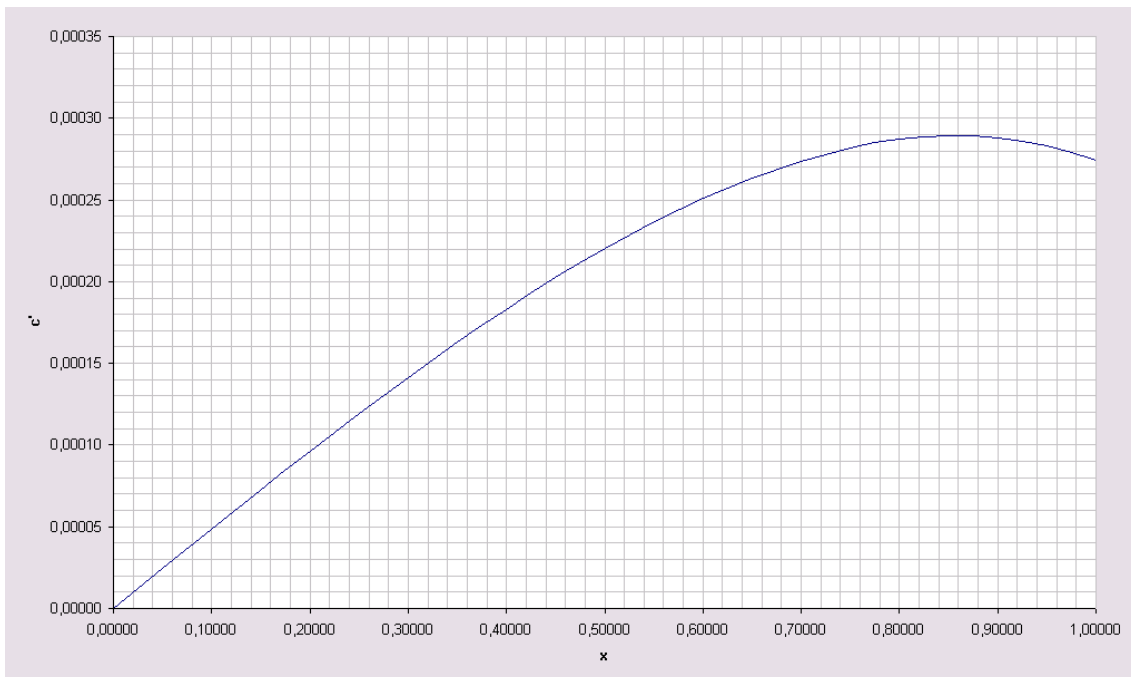


Figure 13: Full model: Concentration profile of c_{PLC}^e .

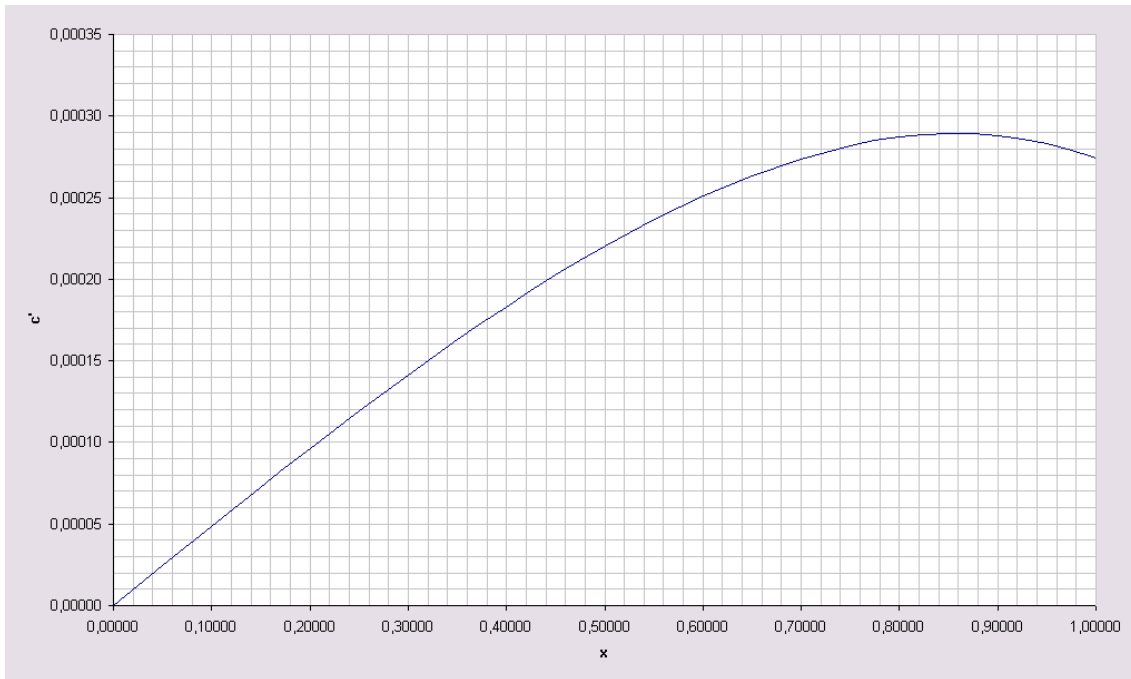


Figure 14: Full model: Profile of $\partial_x c_{PI4,5-P_2}^e$.

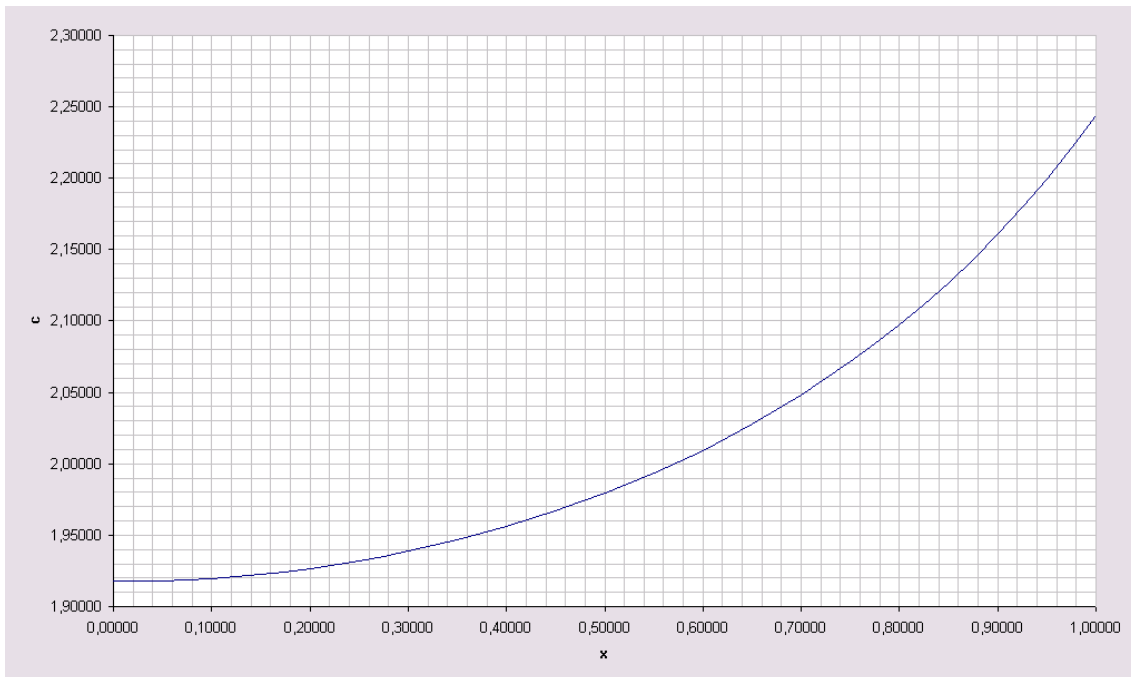


Figure 15: Full model: Concentration profile of $c_{PI4,5-P_2}^m$.

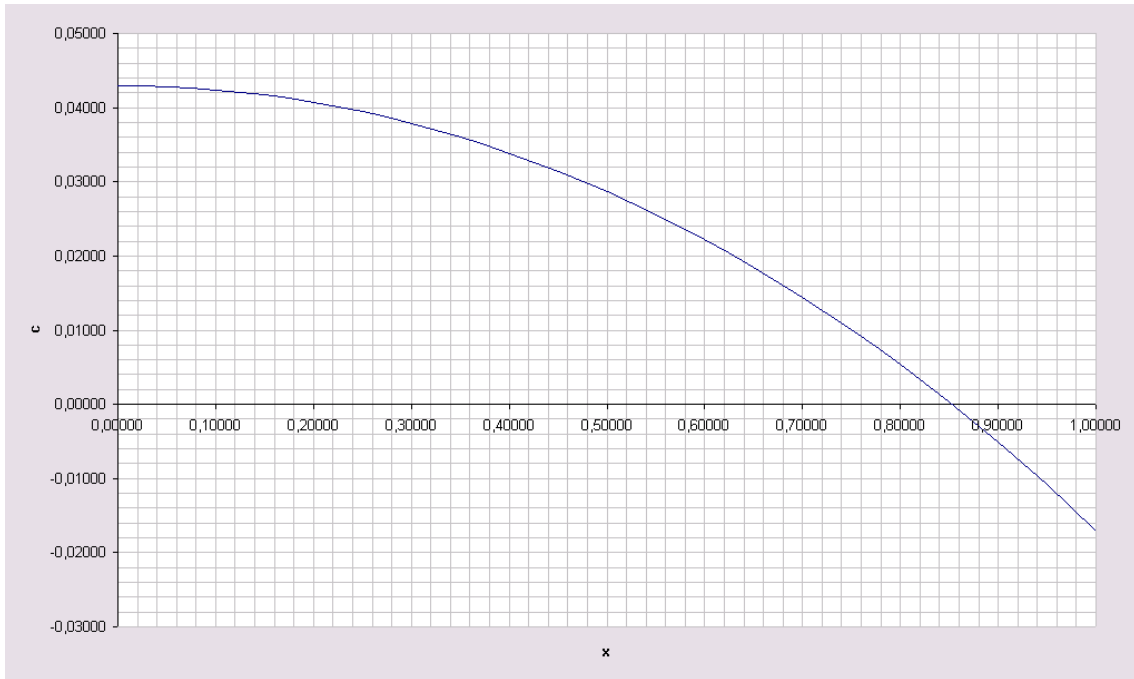


Figure 16: Full model: Concentration profile of c_{IP3}^c .

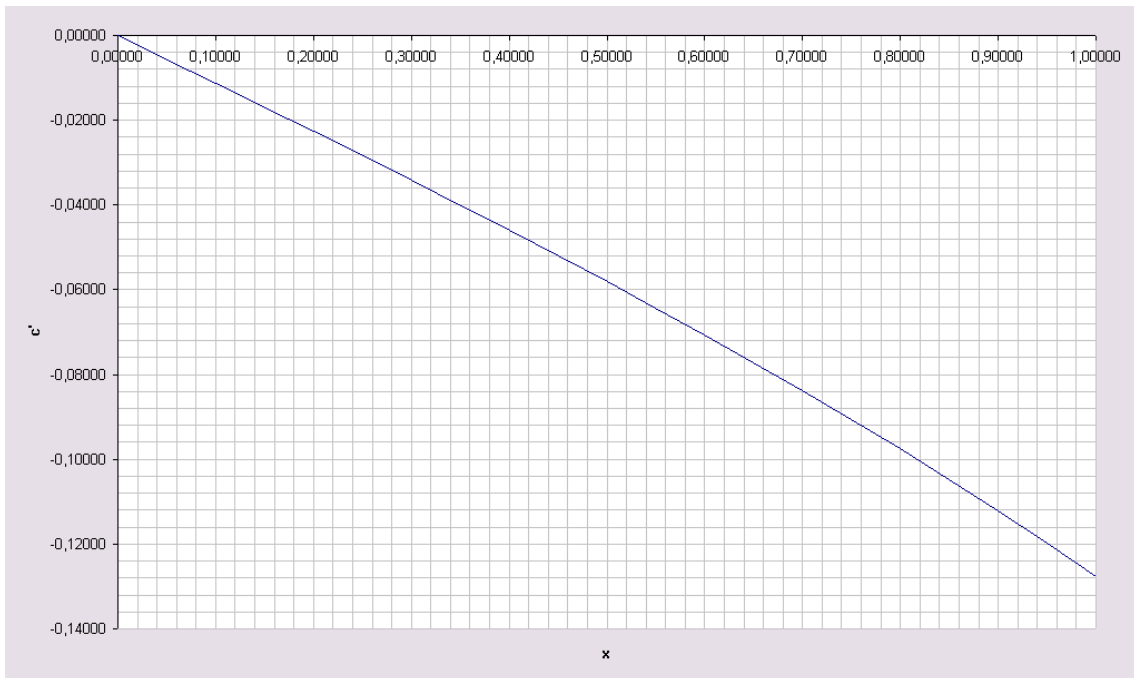


Figure 17: Full model: Profile of $\partial_x c_{IP3}^c$.

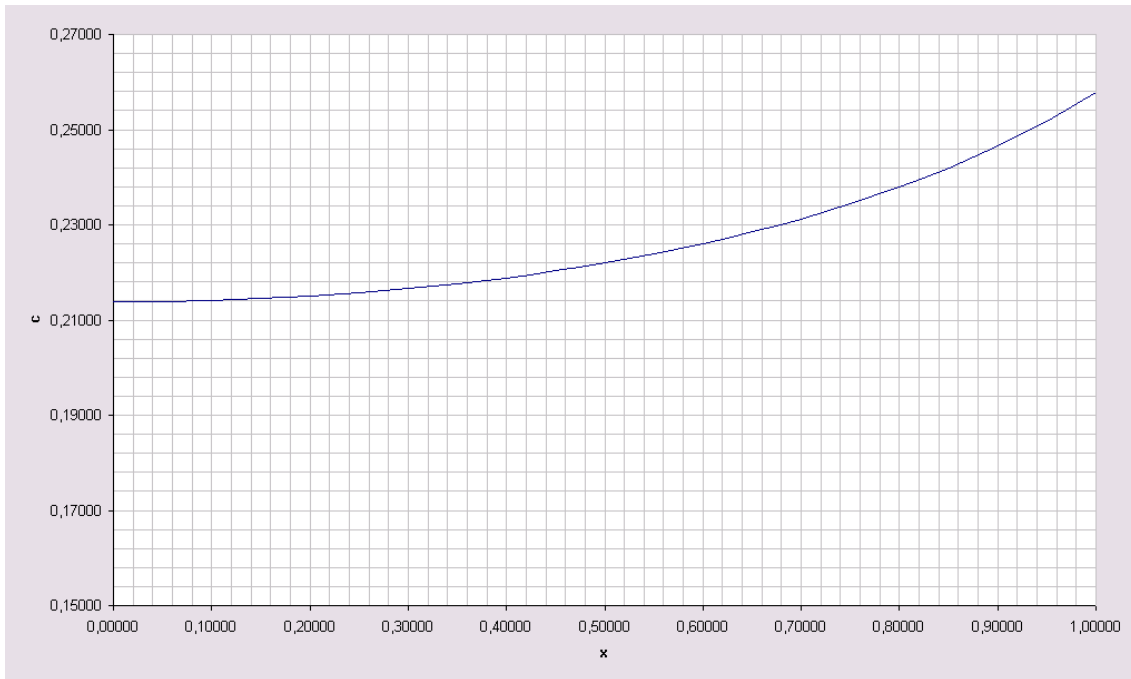


Figure 18: Full model: Concentration profile of $c_{DAG/GAP}^m$.

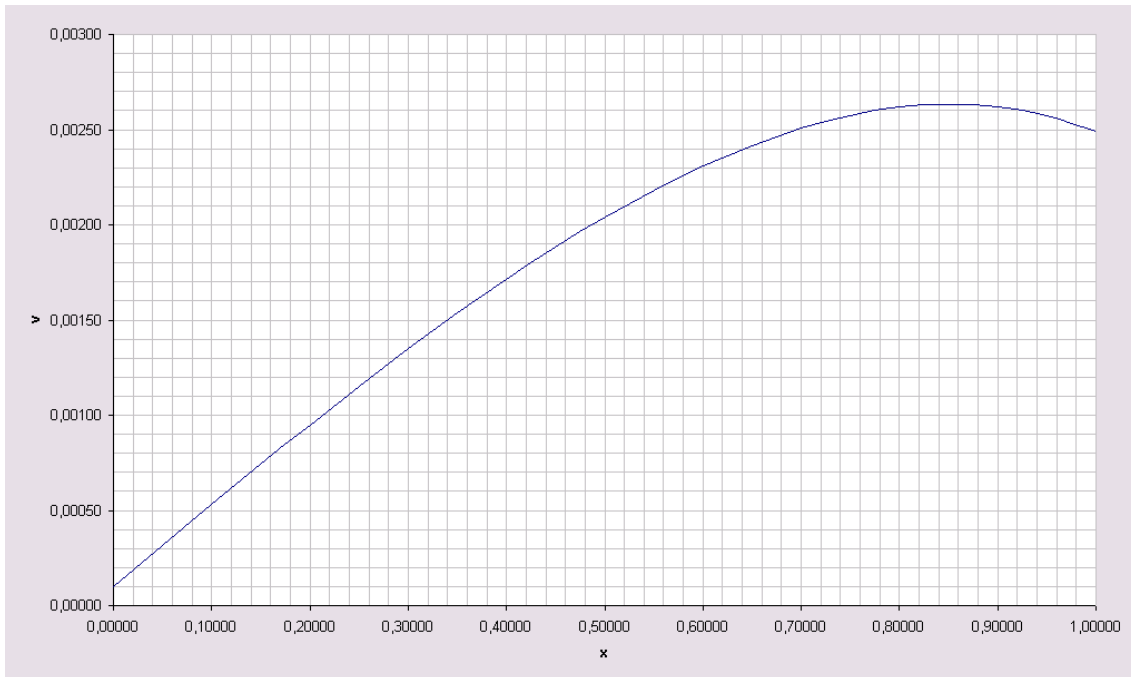


Figure 19: Full model: Velocity profile.

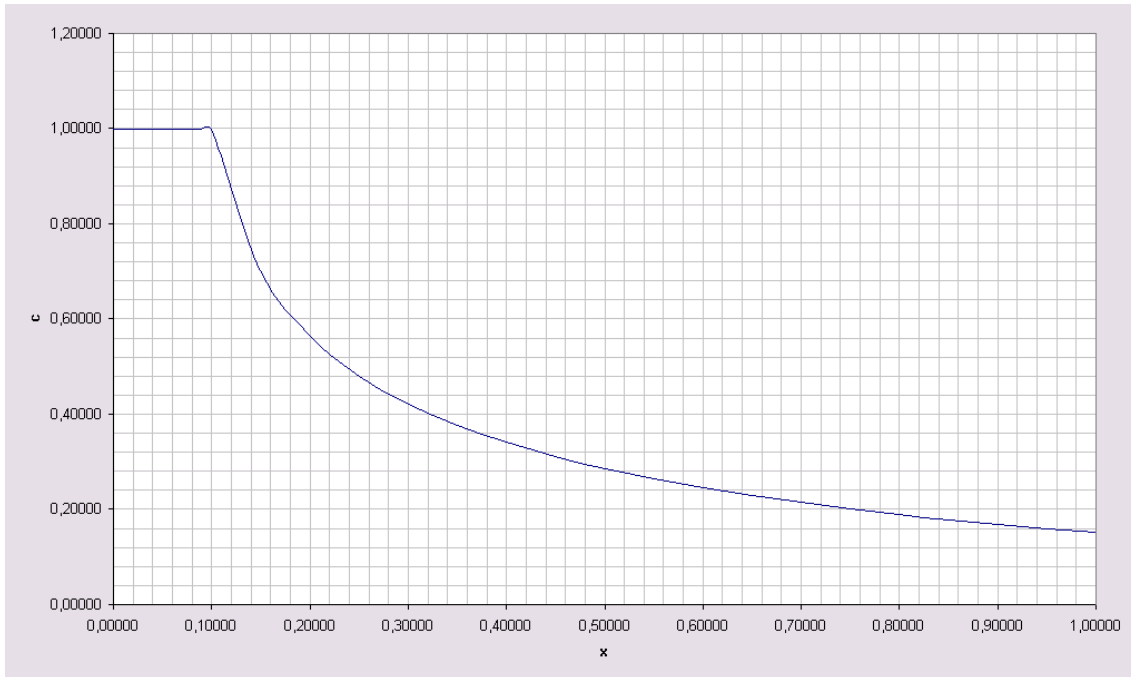


Figure 20: Simplified model: Concentration profile of $c_{\text{Rac-GTP}}^m$.

- $c_{\text{Rac-GTP}}^m$ is at a constant concentration from 0 to 0.1 and rises exponentially afterwards. Simultaneously, the concentration of $c_{\text{Rac-GDP}}^m$ falls anti-proportional to the concentration of $c_{\text{Rac-GTP}}^m$.
- The same results are given for c_{PLC}^m and $c_{\text{PI 4,5-P}_2}^m$: while the concentration of c_{PLC}^m rises, the concentration of $c_{\text{PI 4,5-P}_2}^m$ drops.

5 Conclusion and Outlook

As we described above, the results gained from the simplified model are very satisfactory. This leads to the assumption that also the full model is at least qualitatively meaningful, although it was not possible to create results that answer our expectations. That is probably due to the random choice of the parameters. Additionally, we could not set $V = 0$ at $x = 0$, but chose a small value. This derives from the fact that we neglected diffusion processes in the membrane which could influence the results for regions with a small velocity. To verify the correctness of the model further experiments should be processed. Then the model could be fitted to the experimental data and hence the order of magnitude of the parameters could be determined to adjust the model.

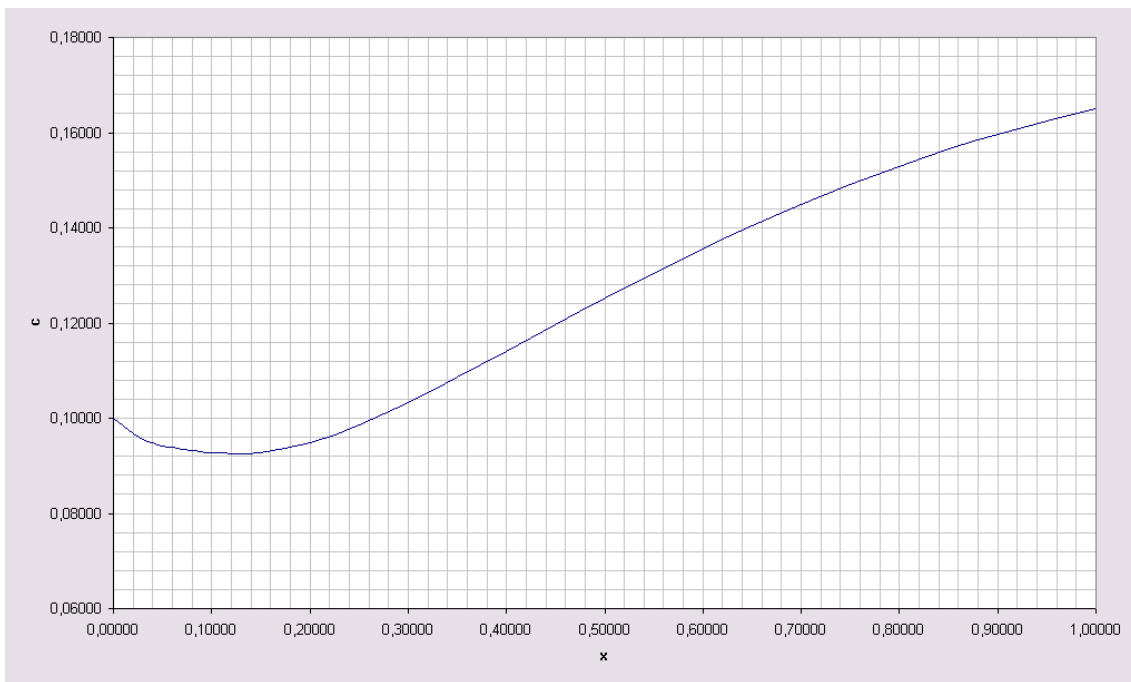


Figure 21: Simplified model: Concentration profile of $c_{\text{Rac-GDP}}^m$.

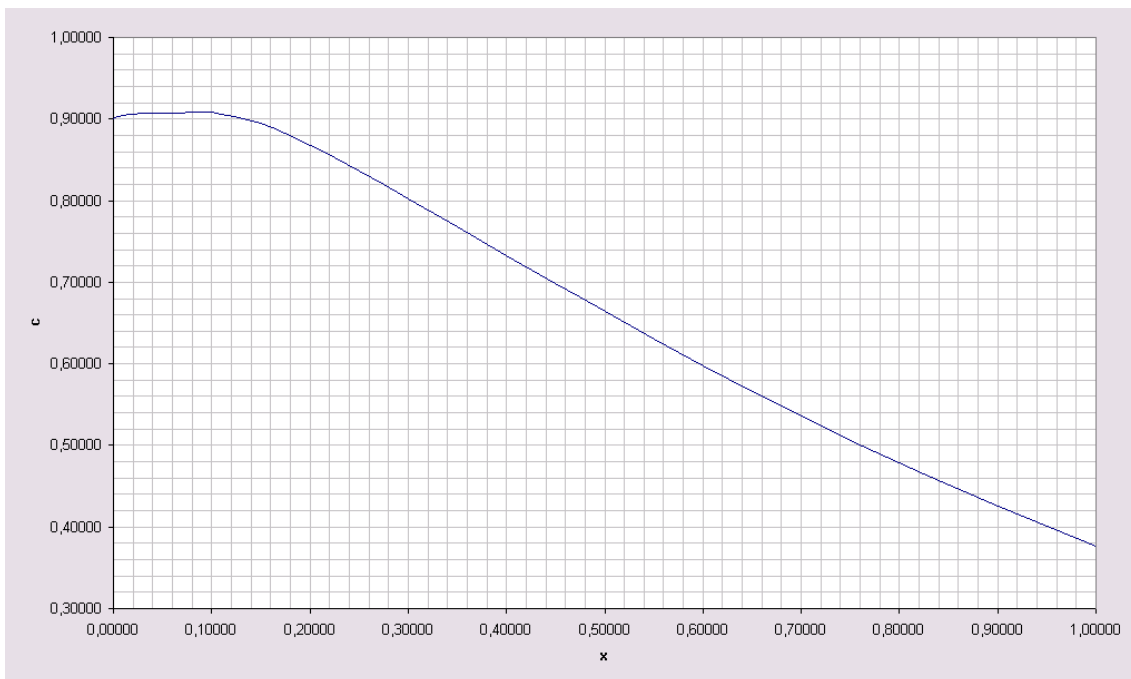


Figure 22: Simplified model: Concentration profile of $c_{\text{PI 4,5-P}_2}^m$.

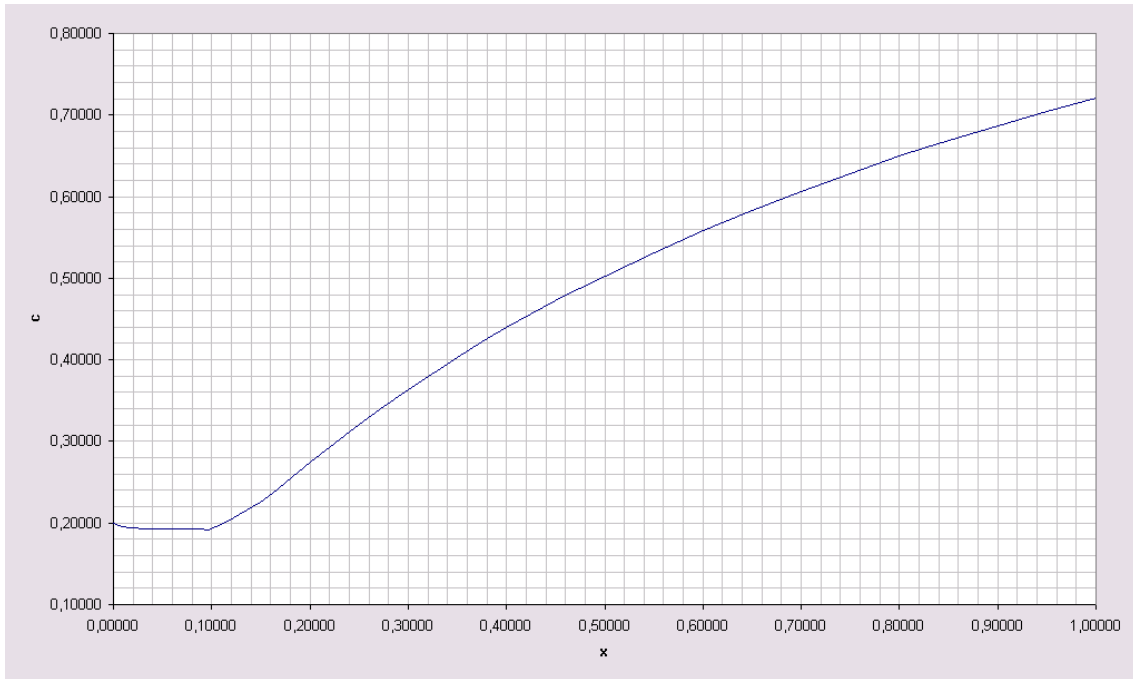


Figure 23: Simplified model: Concentration profile of c_{PLC}^m .

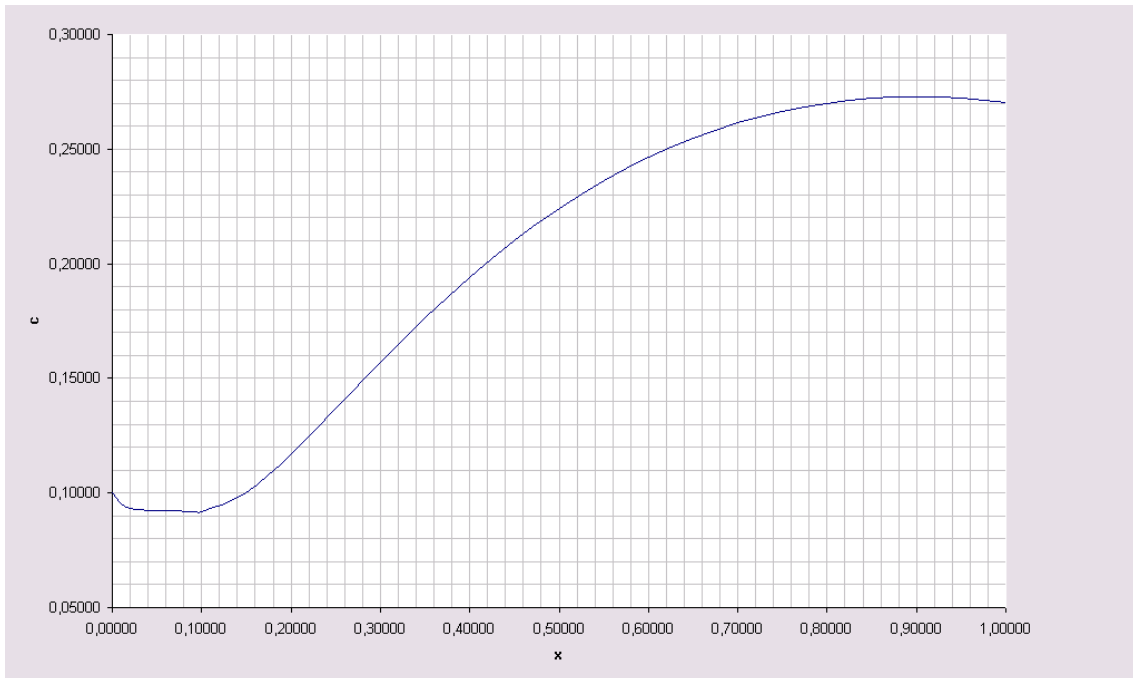


Figure 24: Simplified model: Concentration profile of $c_{DAG/GAP}^m$.

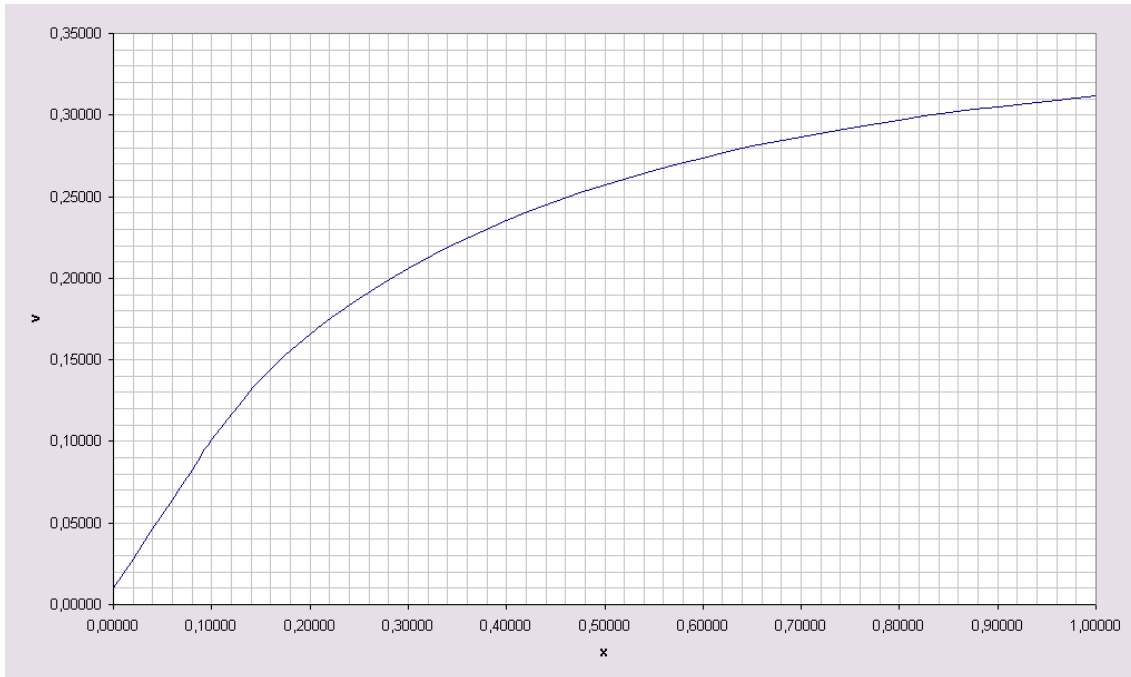


Figure 25: Simplified model: Velocity profile.

6 Appendix

6.1 Program Code of the complete Model implemented in MUSCOD-II

```

/*
 *
 * MUSCOD-II/BIOOPT/SRC/pollentubegrowth.c
 * (c) Johannes Horlemann, 2005
 *
 */

#include <math.h> #include <stdio.h>

#include "def_usrmod.h" #include "def_ind.h"

#define NMOS 1

#define NP 22

#define NRC 0

#define NRCE 0

#define NXD 14

#define NXA 0

```

```

#define NU      0

#define NPR     0

#define NRD_S  10

#define NRDE_S 10

static void ffcn(double *t, double *xd, double *xa, double *u,
    double *p, double *rhs, double *rwh, long *iwh, long *info)
{

double k0,k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12,k13,k14,k15, \
k16,k17,D1,D2,D3,D4,V,Vdot,cmRacGTP,cmRacGDP,ccGDIRacGDP,\
ccGDIRacGDPdot,ccGDI,ccGDI dot,cmPLC,ccPLC,ccPLCdot,cmPIP2,\
ccIP3,ccIP3dot,ccGDIRacGDPdotdot,ccGDI dotdot,ccPLCdotdot,\
ccIP3dotdot,cmRacGTPdot,cmRacGDPdot,cmPLCdot,cmPIP2dot, \
cmDAGGAPdot,cmDAGGAP;

/* Umbenennung der Zustaende */

cmRacGTP=xd[0];
cmRacGDP=xd[1];
ccGDIRacGDP=xd[2];
ccGDIRacGDPdot=xd[3];
ccGDI=xd[4];
ccGDI dot=xd[5];
cmPLC=xd[6];
ccPLC=xd[7];
ccPLCdot=xd[8];
cmPIP2=xd[9];
ccIP3=xd[10];
ccIP3dot=xd[11];
cmDAGGAP=xd[12];
V=xd[13];

/* Parameter */
k0=p[0];
k1=p[1];
k2=p[2];
k3=p[3];
k4=p[4];
k5=p[5];
k6=p[6];
k7=p[7];
k8=p[8];
k9=p[9];
k10=p[10];
k11=p[11];

```

```

k12=p[12];
k13=p[13];
k14=p[14];
k15=p[15];
k16=p[16];
k17=p[17];

D1=p[18];
D2=p[19];
D3=p[20];
D4=p[21];

/* Gleichungen */

Vdot=k0*cmRacGTP*cmPIP2*ccIP3; cmRacGTPdot=(-cmRacGTP*Vdot - (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP + (k5*ccGDIRacGDP))/(V);
cmRacGDPdot=(-cmRacGDP*Vdot + (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP - (k6*ccGDI*cmRacGDP))/(V);
ccGDIRacGDPdotdot=((k5*ccGDIRacGDP) - (k6*ccGDI*cmRacGDP))/D1;
ccGDIidotdot=(-k5*ccGDIRacGDP) + (k6*ccGDI*cmRacGDP)/D2;
cmPLCdot=(-cmPLC*Vdot - k7*cmPLC +
k8*(exp(-k9*cmRacGTP))*ccPLC)/(V); ccPLCdotdot=(-k7*cmPLC +
k8*(exp(-k9*cmRacGTP))*ccPLC)/D3; cmPIP2dot=(-cmPIP2*Vdot +
k10*(cmRacGTP/(k11+k12*cmRacGTP)) -
((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2))/(V);
ccIP3dotdot=(-((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2) +
k16*ccIP3)/D4; cmDAGGAPdot=(-cmDAGGAP*Vdot +
((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2) - k17*cmDAGGAP)/(V);

/* Output (Gleiche Reihenfolge wie Zustaende) */

rhs[0]=cmRacGTPdot; rhs[1]=cmRacGDPdot; rhs[2]=ccGDIRacGDPdot;
rhs[3]=ccGDIRacGDPdotdot; rhs[4]=ccGDIidot; rhs[5]=ccGDIidotdot;
rhs[6]=cmPLCdot; rhs[7]=ccPLCdot; rhs[8]=ccPLCdotdot;
rhs[9]=cmPIP2dot; rhs[10]=ccIP3dot; rhs[11]=ccIP3dotdot;
rhs[12]=cmDAGGAPdot; rhs[13]=Vdot; }

static void rfcns(
    double *ts, /* physical time (I) */
    double *xd, /* differential state vector (I) */
    double *sa, /* algebraic state vector (I) */
    double *u, /* control vector (I) */
    double *p, /* global model parameter vector (I) */
    double *pr, /* local i.p.c. parameter vector (I) */
    double *res, /* i.p.c. residual (0) */
    long *dpnd, /* argument dependency code (I/0) */
    long *info /* error code (0) */
) /*
* decoupled or coupled interior point constraint (i.p.c.) residual

```

```

*/
{

double k0,k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12,k13,k14,k15, \
k16,k17,D1,D2,D3,D4,V,Vdot,cmRacGTP,cmRacGDP,ccGDIRacGDP,\
ccGDIRacGDPdot,ccGDI,ccGDIIdot,cmPLC,ccPLC,ccPLCdot,cmPIP2,\
ccIP3,ccIP3dot,ccGDIRacGDPdotdot,ccGDIIdotdot,ccPLCdotdot,\
ccIP3dotdot,cmRacGTPdot,cmRacGDPdot,cmPLCdot,cmPIP2dot, \
cmDAGGAPdot,cmDAGGAP;

    if (*dpnd) {
        *dpnd = RFCN_DPND(*ts, *xd, *sa, *u, *p, *pr);
        return;
    }

/* Umbenennung der Zustaende */

cmRacGTP=xd[0];
cmRacGDP=xd[1];
ccGDIRacGDP=xd[2];
ccGDIRacGDPdot=xd[3];
ccGDI=xd[4];
ccGDIIdot=xd[5];
cmPLC=xd[6];
ccPLC=xd[7];
ccPLCdot=xd[8];
cmPIP2=xd[9];
ccIP3=xd[10];
ccIP3dot=xd[11];
cmDAGGAP=xd[12];
V=xd[13];

/* Parameter */
k0=p[0];
k1=p[1];
k2=p[2];
k3=p[3];
k4=p[4];
k5=p[5];
k6=p[6];
k7=p[7];
k8=p[8];
k9=p[9];
k10=p[10];
k11=p[11];
k12=p[12];
k13=p[13];
k14=p[14];

```

```

k15=p[15];
k16=p[16];
k17=p[17];

D1=p[18];
D2=p[19];
D3=p[20];
D4=p[21];

/* Gleichungen */

Vdot=k0*cmRacGTP*cmPIP2*ccIP3; cmRacGTPdot=(-cmRacGTP*Vdot - (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP + (k5*ccGDIRacGDP))/(V);
cmRacGDPdot=(-cmRacGDP*Vdot + (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP - (k6*ccGDI*cmRacGDP))/(V);
ccGDIRacGDPdotdot=((k5*ccGDIRacGDP) - (k6*ccGDI*cmRacGDP))/D1;
ccGDIidotdot=(-k5*ccGDIRacGDP) + (k6*ccGDI*cmRacGDP)/D2;
cmPLCdot=(-cmPLC*Vdot - k7*cmPLC +
k8*(exp(-k9*cmRacGTP))*ccPLC)/(V); ccPLCdotdot=(-k7*cmPLC +
k8*(exp(-k9*cmRacGTP))*ccPLC)/D3; cmPIP2dot=(-cmPIP2*Vdot +
k10*(cmRacGTP/(k11+k12*cmRacGTP)) -
((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2))/(V);
ccIP3dotdot=(-((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2) +
k16*ccIP3)/D4; cmDAGGAPdot=(-cmDAGGAP*Vdot +
((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2) - k17*cmDAGGAP)/(V);

res[0]=cmRacGTPdot; res[1]=cmRacGDPdot; res[2]=ccGDIRacGDPdot;
res[3]=ccGDIidot; res[4]=cmPLCdot; res[5]=ccPLCdot; res[6]=cmPIP2dot;
res[7]=ccIP3dot; res[8]=cmDAGGAPdot; res[9]=V-0.0001; }

void def_model(void) {
    def_mdims(NMOS, NP, NRC, NRCE);

    def_msolver(1, def_DAESOL);
    def_mstage(
        0,
        NXD, NXA, NU,
        NULL, NULL,
        0, 0, 0, NULL, ffcn, NULL,
        NULL, NULL,
        0
    );
    def_mpc(
        0,
        "Start Point",
        NPR,
        NRD_S, NRDE_S,
        rfcns,

```

```

        NULL
    );

}

* *
* MUSCOD-II/BIOOPT/DAT/pollentubegrowth.dat
* (c) Johannes Horlemann 2005
*

* # of multiple shooting intervals on each model stage
nshoot
0: 20

* model stage duration start values, scale factors, and bounds
h
0: 1

h_sca
0: 1

h_min
0: 0

h_max
0: 1

h_fix
0: 1

* specification mode for differential state variable start values
s_spec
2

* differential state start values, scale factors, and bounds
sd_fix(0,0)
0: 0
1: 0
2: 0
3: 0
4: 0
5: 0
6: 0
7: 0
8: 0
9: 0
10: 0
11: 0

```

12: 0
13: 0

sd(0,0)

0: 1.0000000000000001E-001
1: 9.3442723785570669E-001
2: 1.7411825440477019E-001
3: 0.0000000000000000E+000
4: 3.0514667238800169E-001
5: 0.0000000000000000E+000
6: 6.3371589403748707E-002
7: 8.3333529674375309E-002
8: 0.0000000000000000E+000
9: 2.0964083014468726E+000
10: 4.4194745294530616E-002
11: 0.0000000000000000E+000
12: 2.6987274760099655E-001
13: 1.0000000000000001E-001

sd(0,e)

0: 0
1: 0
2: 0
3: 0
4: 0
5: 0
6: 0
7: 0
8: 0
9: 0
10: 0
11: 0
12: 0
13: 0

sd_sca(*,*)

0: 1
1: 1
2: 1
3: 1
4: 1
5: 1
6: 1
7: 1
8: 1
9: 1
10: 1
11: 1
12: 1

13: 1

sd_min(*,*)

0: -10e+20
1: -10e+20
2: -10e+20
3: -10e+20
4: -10e+20
5: -10e+20
6: -10e+20
7: -10e+20
8: -10e+20
9: -10e+20
10: 0
11: -10e+20
12: -10e+20
13: -10e+20

sd_max(*,*)

0: 10e+20
1: 10e+20
2: 10e+20
3: 10e+20
4: 10e+20
5: 10e+20
6: 10e+20
7: 10e+20
8: 10e+20
9: 10e+20
10: 10e+20
11: 10e+20
12: 10e+20
13: 10e+20

xd_name

0: cmRacGTP
1: cmRacGDP
2: ccGDIRacGDP
3: ccGDIRacGDPdot
4: ccGDI
5: ccGDI dot
6: >cmPLC
7: ccPLC
8: ccPLCdot
9: cmPIP2
10: ccIP3
11: ccIP3dot
12: ccDAG
13: V

p
0: 0.5
1: 0.5
2: 0.1
3: 1
4: 0
5: 0.3
6: 0.2
7: 0.5
8: 0.4
9: 0.6
10: 0.7
11: 1
12: 0
13: 0.6
14: 1
15: 0
16: 0.2
17: 0.3
18: 0.5
19: 0.5
20: 0.5
21: 0.5

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

p_sca

0: 0.0001
1: 0.0001
2: 0.0001
3: 0.0001
4: 0.0001
5: 0.0001
6: 0.0001
7: 0.0001
8: 0.0001
9: 0.0001
10: 0.0001
11: 0.000
12: 0.0001
13: 0.0001
14: 0.0001
15: 0.0001
16: 0.0001
17: 0.0001
18: 0.0001
19: 0.0001
20: 0.0001
21: 0.0001

p_min

0: -100
1: -100
2: -100
3: -100
4: -100
5: -100
6: -100
7: -100
8: -100
9: -100
10: -100
11: -100
12: -100
13: -100
14: -100
15: -100
16: -100
17: -100
18: -100
19: -100
20: -100
21: -100

p_max

0: 100
1: 100
2: 100
3: 100
4: 100
5: 100
6: 100
7: 100
8: 100
9: 100
10: 100
11: 100
12: 100
13: 100
14: 100
15: 100
16: 100
17: 100
18: 100
19: 100
20: 100
21: 100

p_name

0: !k0
1: !k1
2: !k2
3: !k3
4: !k4
5: !k5
6: !k6
7: !k7
8: !k8
9: !k9
10: !k10
11: !k11
12: !k12
13: !k13
14: !k14
15: !k15
16: !k16
17: !k17
18: !D1
19: !D2
20: !D3
21: !D4

h_name

0: !

```
rd_sca(*,*)
```

```
0: 1
```

```
1: 1
```

```
2: 1
```

```
3: 1
```

```
4: 1
```

```
5: 1
```

```
6: 1
```

```
7: 1
```

```
8: 1
```

```
9: 1
```

```
10: 1
```

```
11: 1
```

```
12: 1
```

```
13: 1
```

```
of_sca
```

```
1
```

```
of_min
```

```
-2
```

```
of_max
```

```
0
```

```
of_name
```

```
nhist 20
```

6.2 Program Code of the simplified Model implemented in MUSCOD-II

```
/*  
 *  
 * MUSCOD-II/BIOOPT/SRC/pollentubegrowtheinfach.c  
 * (c) Johannes Horlemann, 2005  
 *  
 */
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```
#include "def_usrmod.h"
```

```
#include "def_ind.h"
```

```

#define NMOS 2

#define NP 21

#define NRC 0

#define NRCE 0

#define NXD 6

#define NXA 0

#define NU 0

#define NPR 0

#define NRD_S 0

#define NRDE_S 0

static void ffcn(double *t, double *xd, double *xa, double *u,
    double *p, double *rhs, double *rwh, long *iwh, long *info, long k5inhibit)
{

double k0,k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12,k13,k14,k15,\
k16,V,Vdot,cmRacGTP,cmRacGDP,ccGDIRacGDP,ccGDI,cmPLC,ccPLC,\
cmPIP2,ccIP3,cmRacGTPdot,cmRacGDPdot,cmPLCdot,cmPIP2dot, \
cmDAGGAPdot,cmDAGGAP;

/* Umbenennung der Zustaende */

cmRacGTP=xd[0];
cmRacGDP=xd[1];
cmPLC=xd[2];
cmPIP2=xd[3];
cmDAGGAP=xd[4];
V=xd[5];

/* Parameter */
k0=p[0];
k1=p[1];
k2=p[2];
k3=p[3];
k4=p[4];
k5=p[5];
k6=p[6];
k7=p[7];
k8=p[8];
k9=p[9];

```

```

k10=p[10];
k11=p[11];
k12=p[12];
k13=p[13];
k14=p[14];
k15=p[15];
k16=p[16];

ccGDIRacGDP=p[17];
ccGDI=p[18];
ccPLC=p[19];
ccIP3=p[20];

/* Gleichungen */

Vdot=k0*cmRacGTP*cmPIP2*ccIP3; cmRacGTPdot=(-cmRacGTP*Vdot - (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP +
(k5inhibit*k5*ccGDIRacGDP))/(V); cmRacGDPdot=(-cmRacGDP*Vdot + (k1 +
k2*(cmDAGGAP/(k3+k4*cmDAGGAP)))*cmRacGTP - (k6*ccGDI*cmRacGDP))/(V);
cmPLCdot=(-cmPLC*Vdot - k7*cmPLC +
k8*(exp(-k9*cmRacGTP))*ccPLC)/(V); cmPIP2dot=(-cmPIP2*Vdot +
k10*(cmRacGTP/(k11+k12*cmRacGTP)) -
((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2))/(V);
cmDAGGAPdot=(-cmDAGGAP*Vdot + ((k13*(cmPLC/(k14+k15*cmPLC)))*cmPIP2)
- k16*cmDAGGAP)/(V);

/* Output (Gleiche Reihenfolge wie Zustaende) */

rhs[0]=cmRacGTPdot; rhs[1]=cmRacGDPdot; rhs[2]=cmPLCdot;
rhs[3]=cmPIP2dot; rhs[4]=cmDAGGAPdot; rhs[5]=Vdot; }

static void ffcn0(double *t, double *xd, double *xa, double *u,
double *p, double *rhs, double *rwh, long *iwh, long *info)
{
    ffcn(t,xd,xa,u,p,rhs,rwh,iwh,info, 1);
}

static void ffcn1(double *t, double *xd, double *xa, double *u,
double *p, double *rhs, double *rwh, long *iwh, long *info)
{
    ffcn(t,xd,xa,u,p,rhs,rwh,iwh,info, 0);
}

void def_model(void) {
    def_mdims(NMOS, NP, NRC, NRCE);
}

```

```

def_msolver(1, def_DAESOL);
def_mstage(
    0,
    NXD, NXA, NU,
    NULL, NULL,
    0, 0, 0, NULL, ffcn0, NULL,
    NULL, NULL,
    0
);
def_mstage(
    1,
    NXD, NXA, NU,
    NULL, NULL,
    0, 0, 0, NULL, ffcn1, NULL,
    NULL, NULL,
    0
);
}

```

```
* *
```

```
* MUSCOD-II/BIOOPT/DAT/pollentubegrowtheinfach.dat
```

```
* (c) Johannes Horlemann 2005
```

```
*
```

```
* # of multiple shooting intervals on each model stage
```

```
nshoot
```

```
0: 10
```

```
1: 18
```

```
* model stage duration start values, scale factors, and bounds
```

```
h
```

```
0: 0.1
```

```
1: 0.9
```

```
h_sca
```

```
0: 1
```

```
1: 1
```

```
h_min
```

```
0: 0
```

```
1: 0.1
```

```
h_max
```

```
0: 1
```

```
1: 1
```

```
h_fix
```

```

0: 1
1: 1

* specification mode for differential state variable start values
s_spec
2

* differential state start values, scale factors, and bounds
sd_fix(0,0)
0: 0
1: 0
2: 0
3: 0
4: 0
5: 0

sd(0,0)
0: 1
1: 0.1
2: 0.2
3: 0.9
4: 0.1
5: 0.01

sd(0,e)
0: 0
1: 0
2: 0
3: 0
4: 0
5: 0

sd_sca(*,*)
0: 1
1: 1
2: 1
3: 1
4: 1
5: 1

sd_min(*,*)
0: -10e+20
1: -10e+20
2: -10e+20
3: -10e+20
4: -10e+20
5: -10e+20

sd_max(*,*)

```

0: 10e+20
1: 10e+20
2: 10e+20
3: 10e+20
4: 10e+20
5: 10e+20

xd_name

0: cmRacGTP
1: cmRacGDP
2: >cmPLC
3: cmPIP2
4: ccDAGGAP
5: V

p

0: 1
1: 0
2: 1
3: 1
4: 0
5: 1
6: 0.1
7: 1
8: 1
9: 1
10: 1
11: 1
12: 0
13: 1
14: 1
15: 0
16: 1
17: 1
18: 1
19: 1
20: 1

p_fix

0: 1
1: 1
2: 1
3: 1
4: 1
5: 1
6: 1
7: 1
8: 1
9: 1

10: 1
11: 1
12: 1
13: 1
14: 1
15: 1
16: 1
17: 1
18: 1
19: 1
20: 1

p_sca

0: 0.0001
1: 0.0001
2: 0.0001
3: 0.0001
4: 0.0001
5: 0.0001
6: 0.0001
7: 0.0001
8: 0.0001
9: 0.0001
10: 0.0001
11: 0.0001
12: 0.0001
13: 0.0001
14: 0.0001
15: 0.0001
16: 0.0001
17: 0.0001
18: 0.0001
19: 0.0001
20: 0.0001

p_min

0: -100
1: -100
2: -100
3: -100
4: -100
5: -100
6: -100
7: -100
8: -100
9: -100
10: -100
11: -100
12: -100

13: -100
14: -100
15: -100
16: -100
17: -100
18: -100
19: -100
20: -100

p_max

0: 100
1: 100
2: 100
3: 100
4: 100
5: 100
6: 100
7: 100
8: 100
9: 100
10: 100
11: 100
12: 100
13: 100
14: 100
15: 100
16: 100
17: 100
18: 100
19: 100
20: 100

p_name

0: !k0
1: !k1
2: !k2
3: !k3
4: !k4
5: !k5
6: !k6
7: !k7
8: !k8
9: !k9
10: !k10
11: !k11
12: !k12
13: !k13
14: !k14
15: !k15

```
16: !k16
17: !ccGDIRacGDP
18: !ccGDI
19: !ccPLC
20: !ccIP3
```

```
h_name
0: !
1:!
```

```
rd_sca(*,*)
0: 1
1: 1
2: 1
3: 1
4: 1
5: 1
6: 1
7: 1
8: 1
9: 1
10: 1
11: 1
12: 1
13: 1
```

```
of_sca
1
```

```
of_min
-2
```

```
of_max
0
```

```
of_name
```

```
nhist
20
```

6.3 Program Code of the simplified Model implemented in Matlab 6.5

```
function dydx = pt(x,y)
```

```
% Constants used in the model
```

```
k_0 = 1.00;
k_1 = 0.00; k_2 = 1.00; k_3 = 1.00; k_4 = 0.00;
```

```

k_5 = 1.00;
k_6 = 0.10;
k_7 = 1.00;
k_8 = 1.00;
k_9 = 1.00;
k_10 = 1.00; k_11 = 1.00; k_12 = 0.00;
k_13 = 1.00; k_14 = 1.00; k_15 = 0.00;
k_16 = 1.00;

% A more readable form of the kinetics
V = y( 1);
mRacGTP = y( 2);
mRacGDP = y( 3);
mPLC = y( 4);
mPI45P2 = y( 5);
mDAGGAP = y( 6);
cGDIRacGDP = 1.0;
cGDI = 1.0;
cPLC = 1.0;

% Definitions of the kinetics
KV = + k_0 * mRacGTP * mPI45P2;
KmRacGTP = - (k_1 + k_2*mDAGGAP/(k_3+k_4*mDAGGAP)) * mRacGTP
+ 1*k_5 *(x<0.1)* cGDIRacGDP;
KmRacGDP = + (k_1 + k_2*mDAGGAP/(k_3+k_4*mDAGGAP)) * mRacGTP
- k_6 * cGDI * mRacGDP;
KmPLC = - k_7 * mPLC + (k_8*exp(-k_9*mRacGTP)) * cPLC;
KmPI45P2 = + (k_10*mRacGTP/(k_11+k_12*mRacGTP))
- (k_13*mPLC/(k_14+k_15*mPLC)) * mPI45P2;
KmDAGGAP = + (k_13*mPLC/(k_14+k_15*mPLC)) * mPI45P2
- k_16 * mDAGGAP;

% Setting up the the right hand side
R_V = KV;
R_mRacGTP = 1/V * ( - mRacGTP*R_V + KmRacGTP);
R_mRacGDP = 1/V * ( - mRacGDP*R_V + KmRacGDP);
R_mPLC = 1/V * ( - mPLC*R_V + KmPLC);
R_mPI45P2 = 1/V * ( - mPI45P2*R_V + KmPI45P2);
R_mDAGGAP = 1/V * ( - mDAGGAP*R_V + KmDAGGAP);

dydx( 1) = R_V;
dydx( 2) = R_mRacGTP;
dydx( 3) = R_mRacGDP;
dydx( 4) = R_mPLC;
dydx( 5) = R_mPI45P2;
dydx( 6) = R_mDAGGAP;

dydx = dydx';

```

```

%Intervall on which we solve the problem
x0 = [0,1];

%Initial Conditions
y0( 1) = + 0.01;           %V
y0( 2) = + 1.00;          %mRacGTP
y0( 3) = + 0.10;          %mRacGDP
y0( 4) = + 0.20;          %mPLC
y0( 5) = + 0.90;          %mPI45P2
y0( 6) = + 0.10;          %mDAGGAP

%The ode-solver
[X,Y] = ode45(@ptneu,x0,y0);

%Plot the resultls
figure; plot(X,Y(:,1));
figure; plot(X,Y(:,2));
figure; plot(X,Y(:,3));
figure; plot(X,Y(:,4));
figure; plot(X,Y(:,5));
figure; plot(X,Y(:,6));

```

Acknowledgments

Thanks to Dirk Hartmann and Moritz Diehl for their support in creating the model and the simulations.

A special thanks to Stefan Quint, Sabine Oberhansl, and Dirk Hartmann for proofreading this report.

References

- [1] Zheng et al., "The Rop GTPase switch turns on polar growth in pollen", Trends in Plant Science, Volume 5, No. 7:298-303, 2000.
- [2] Hepler et al., "Polarized Cell Growth in Higher Plants", Annu. Rev. Cell Dev. Biol., 17:159-87, 2001.
- [3] Kost, Benedikt, "Rac/Rop GTPase signaling in pollen tubes", Power-point presentation, Development Biology, Heidelberg Institute of Plant Science (HIP), 2004.
- [4] Crank, John, "The Mathematics of Diffusion", 2nd Edition, Oxford University Press, 1975.
- [5] Diehl, Moritz, Leineweber, Daniel B., Schfer, Andreas A. S., MUSCOD-II (Release 2.1), Interdisciplinary center of scientific computing (IWR), University of Heidelberg, 2004.
- [6] The MathWorks, Inc., Matlab 6.5 (Release 13), 2002.