Feedback and reversibility in substrate-enzyme reactions as discrete event models

Systems Engineering Group
Department of Mechanical Engineering
Eindhoven University of Technology
PO Box 513
5600 MB Eindhoven
The Netherlands
http://se.wtb.tue.nl/

SE Report: Nr. 2010-01

Feedback and reversibility in substrate-enzyme reactions as discrete event models

D.A.J. van Zwieten, J.E. Rooda,
D. Armbruster and J.D. Nagy

February 12, 2010

ISSN: 1872-1567

SE Report: Nr. 2010-01
Eindhoven, February 2010
SE Reports are available via http://se.wtb.tue.nl/sereports

Abstract

A different approach in modeling reactions between substrate molecules and enzymes is presented in this report. The reactions are modeled using a discrete event model (DEM), mostly used in manufacturing or queueing systems. The DEM has been validated with the most used approach to modeling substrate-enzyme reactions, a set of ordinary differential equations (ODEs). The substrate-enzyme model is extended with feedback of substrate and/or product molecules on the enzyme, resulting in inhibition or activation of the enzyme. A reversible reaction, where the enzyme can react with both the substrate and the product, is also modeled as a DEM and validated with an ODE model. The substrate-enzyme reaction models with feedback is extended to a steady-state system by adding a generator, convertor and exit. Stability of the steady-state system is analyzed and resulted in three distinct regions.

Contents

Chapter 1

Introduction

In biology, millions of substrate-enzyme reactions occur in every living cell. Enzymes are the proteins that catalyze chemical reactions. In enzymatic reactions, the molecules at the beginning of the process are called substrates, and the enzyme reconfigures them into different molecules, called products. Usually these reactions and networks of these reactions are modeled by Ordinary Differential Equations (ODEs). In [7], a start in modeling substrate-enzyme reactions as discrete event models (DEMs) has been presented. In this report we follow this approach, which is mostly used in manufacturing and queueing models, and model different kinds of substrate-enzyme reactions.

The general substrate-enzyme reaction is modeled and simulated with ordinary differential equations (ODEs) in Chapter 2. Chapter 3 gives a short introduction of discrete event models, the stochastic processes and specifies the model parameters before modeling the substrate-enzyme reaction. This model is verified with the ODE model results. Chapter 4 presents an extension of the DEM model with downstream and/or upstream feedback molecules affecting the search process. In Chapter 5 feedback is modeled affecting the reconfiguration process. A reversible reaction is modeled and simulated in Chapter 6. The substrate-enzyme reaction models with feedback and reversibility are used to model and analyze the first steps of the EMP-pathway, the most common type of glycolysis in Chapter 7. This report ends with conclusions and recommendations in Chapter 8.

Chapter 2

ODE approach

Often biological systems are modeled using Ordinary Differential Equations (ODEs). Biological ODE models are continuous deterministic models in which variables represent concentrations of the molecules involved. Models are based on average reaction rates, measured from experiments which relate to the change of concentrations over time. With the reaction equation, initial amounts of the concentrations and the rate of every reaction a model can be constructed.
The general substrate-enzyme reaction scheme is as follows:

$$S + E \leftrightarrow C \rightarrow P + E. \tag{2.1}$$

This scheme describes how enzyme E catalyzes a reaction wherein substrate S is converted via complex C into product P. This reaction takes place in a fixed volume which is assumed to be well mixed and in thermal equilibrium. This reaction scheme can be separated into the following three reactions:

$$S + E \xrightarrow{k_1} C, \tag{2.2a}$$

$$C \xrightarrow{k_2} S + E, \tag{2.2b}$$

$$C \xrightarrow{k_3} P + E. \tag{2.2c}$$

When substrate S and enzyme E collide substrate-enzyme complex C is formed which consists of substrate S bound to enzyme E, see (2.2a). This reaction is characterized by mass-action rate constant k_1 [min^{-1}] and denotes the reaction speed of the search of the substrate and the enzyme for each other. The complex can undergo two different reactions. Complex C can disintegrate into enzyme E and substrate S again, see (2.2b). Rate constant k_2 denotes the failure speed to reconfigure the substrate after forming complex C. In the third reaction, see (2.2c), bounded substrate S in complex C is reconfigured into product P, whereafter complex C falls apart in product P and enzyme E. Rate constant k_3 denotes successful reconfiguration speed. The dynamics of the

system can be written as the following set of ODEs:

$$\frac{dC_\mathrm{S}}{dt} = k_2 C_\mathrm{C} - k_1 C_\mathrm{S} C_\mathrm{E}, \tag{2.3a}$$

$$\frac{dC_\mathrm{E}}{dt} = (k_2 + k_3) C_\mathrm{C} - k_1 C_\mathrm{S} C_\mathrm{E}, \tag{2.3b}$$

$$\frac{dC_\mathrm{C}}{dt} = k_1 C_\mathrm{S} C_\mathrm{E} - (k_2 + k_3) C_\mathrm{C}, \tag{2.3c}$$

$$\frac{dC_\mathrm{P}}{dt} = k_3 C_\mathrm{C}, \tag{2.3d}$$

where C_S, C_E, C_C and C_P denote the concentration of respectively substrate S, enzyme E, complex C and product P. Note that enzymes are either unbound represented by symbol E or bound with substrate S to form complex C.

This system has been described by Leonor Michaelis and Maud Menten for the first time in 1913 [6]. They assumed that reconfiguration step k_3, from C to P and E, was the bottleneck and much slower than k_2. Therefore they neglected k_2 in some parts of the model. In [2], Briggs and Haldane proposed that the total concentration of the enzyme is much smaller than the substrate concentration. Since this is typically true for biological reactions, this assumption is usually valid.
Total enzyme concentration $C_{\mathrm{E_t}}$ is equal to the sum of bound and unbound enzyme concentrations:

$$C_{\mathrm{E_t}} = C_\mathrm{E} + C_\mathrm{C}. \tag{2.4}$$

If $C_\mathrm{S} \gg C_{\mathrm{E_t}}$ a steady state is reached in which complex concentration C_C does not change in time:

$$\frac{dC_\mathrm{C}}{dt} = k_1 C_\mathrm{S} C_\mathrm{E} - (k_2 + k_3) C_\mathrm{C} = 0. \tag{2.5}$$

Using (2.4), (2.5) can be reunited as:

$$C_\mathrm{S} C_{\mathrm{E_t}} = C_\mathrm{S} C_\mathrm{C} + \frac{k_2 + k_3}{k_1} C_\mathrm{C}. \tag{2.6}$$

The relation between the rate constants is defined by the Michaelis-Menten constant K_m, see [5]:

$$K_\mathrm{m} = \frac{k_2 + k_3}{k_1}, \tag{2.7}$$

and hence:

$$\frac{dC_\mathrm{P}}{dt} = k_3 C_{\mathrm{E_t}} \frac{C_\mathrm{S}}{K_\mathrm{m} + C_\mathrm{S}}, \tag{2.8}$$

The maximal reconfiguration rate of the reaction V_max is denoted by the rate constant of the product forming step k_3 multiplied by the total enzyme concentration $C_{\mathrm{E_t}}$ in the considered volume V.

$$V_\mathrm{max} = k_3 C_{\mathrm{E_t}} V. \tag{2.9}$$

If unbound enzyme E and complex concentration C do not change in time, the decrease in concentration of substrate S is equal to the increase in concentration of product P. The resulting ODEs for the substrate-enzyme reaction with Michaelis-Menten constant K_m are presented in (2.10).

$$\frac{dC_\mathrm{P}}{dt} = V_\mathrm{max} \frac{C_\mathrm{S}}{K_\mathrm{m} + C_\mathrm{S}}, \tag{2.10a}$$

$$\frac{dC_\mathrm{S}}{dt} = -V_\mathrm{max} \frac{C_\mathrm{S}}{K_\mathrm{m} + C_\mathrm{S}}, \tag{2.10b}$$

This reaction scheme is as follows: substrate molecule S binds with enzyme E, the enzyme reconfigures the substrate molecule into product molecule P and the product

molecule unbinds from the enzyme. This reaction is described by (2.11) and a graphical representation of the substrate-enzyme reaction is presented in Figure 2.1.

$$S + E \rightarrow P + E. \tag{2.11}$$

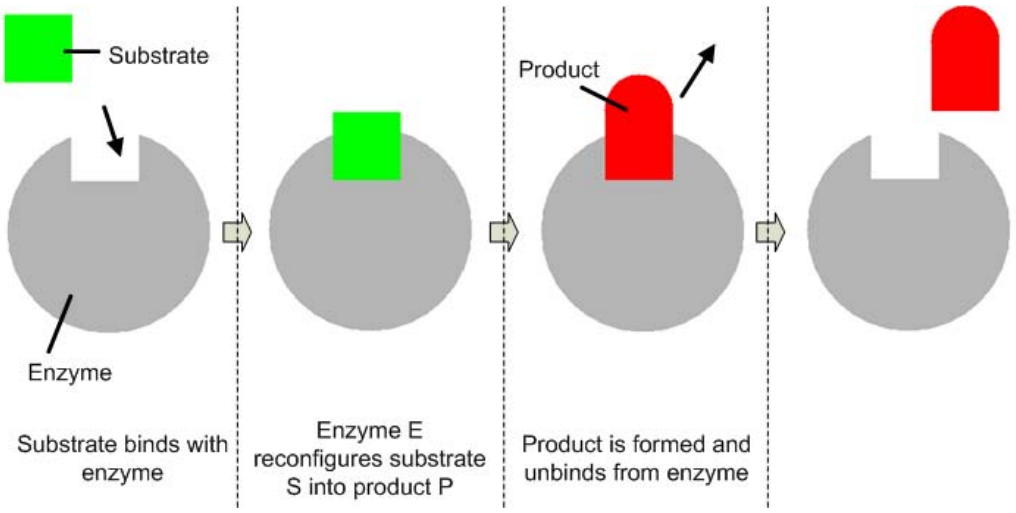

Figure 2.1: Graphical representation of a substrate-enzyme reaction.

The set of ODEs for this system can be numerically solved, given the initial concentrations and the Michealis-Menten constant. In this case, parameters and initial concentrations are used from [9], Chapter 7, see Table A in Appendix A.

Results of this ODE simulation for the typical set of parameters and initial conditions are presented in Figure 2.2. Due to the decreasing concentration of the substrate molecules in this transient simulation, the reaction speed decreases and eventually all substrate molecules are reconfigured into product molecules, as expected.

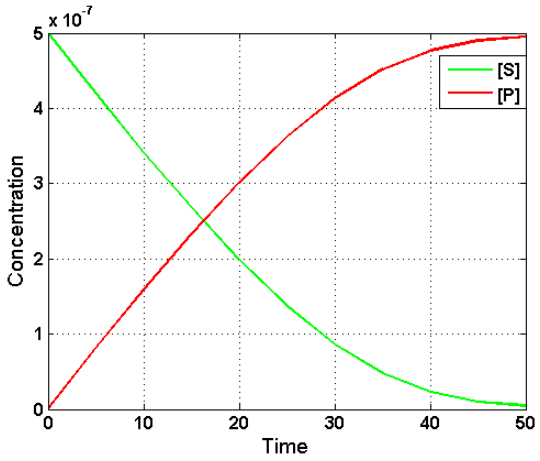

Figure 2.2: Simulation results of the set of ODEs.

Chapter 3

Manufacturing approach

A different approach as he continuous deterministic ODE-models is what we call the "manufacturing" approach. For modeling manufacturing and queueing systems discrete event models (DEMs) are used. A DEM is in discrete time and can contain any given distribution. The DEMs are modeled and simulated using χ [8]. For modeling a substrate-enzyme reaction as a DEM, the reaction can be divided into two distinct processes. The first process, *search* process, describes the search between substrate molecule and enzyme, i.e. the time it takes for a substrate molecule to collide with an enzyme molecule. In the second process, *reconfiguration* process, the enzyme reconfigures the substrate molecule into a product molecule and the product molecule unbinds from the enzyme. Both processes are stochastically independent and, in contrast with ODE models, DEMs can make a distinction between them. To model and simulate the search process, a variable τ_{s} denoting the search time between a substrate molecule and the enzyme is introduced. This variable depends on substrate concentration C_{S}. For high concentrations of substrate the hazard of a collision is large and therefore the search time is small and for low concentrations the collision hazard is small and therefore search time τ_{s} is large. Similar to the search time, a variable denoting the reconfiguration time τ_{r} is introduced for the reconfiguration process. Reconfiguration time τ_{r} corresponds to the process time of the machine. In this manufacturing approach a machine contains a number of enzymes of the same kind. Maximal process rate depends on the number of enzymes of one kind in the considered volume.
A discrete event model representation of the substrate enzyme reaction is presented in Figure 3.1. The concentrations of substrate molecules S and product molecules P are represented by two buffers, respectively B_S and B_P and the reaction process is presented by R_E and depends on the enzyme molecules. From now on we represent reaction processes by circles and buffer processes by squares.

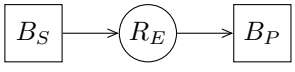

Figure 3.1: Discrete event model representation.

3.1 Parameters

Parameters used in the ODE model are used to obtain parameters for the DEM. With Michaelis-Menten constant K_m given, only search time τ_s and reconfiguration time τ_r have to be calculated. Reconfiguration time is calculated as the minimal processing time (processing at maximal speed V_max):

$$\tau_\mathrm{r} = \frac{1}{V_\mathrm{max}} = \frac{1}{k_3 \cdot C_{\mathrm{E_t}} \cdot V}. \tag{3.1}$$

Total reaction speed of one molecule is denoted by the specific activity of the enzyme v_{enz} multiplied by the number of enzymes. Total reaction time Δt, the sum of search process and reconfiguration process, of one molecule then becomes:

$$\Delta t = \tau_\mathrm{s} + \tau_\mathrm{r} = \frac{1}{v_\mathrm{enz}}. \tag{3.2}$$

Specific activity of the enzyme v_enz is in this case, as presented in(2.10a):

$$v_\mathrm{enz} = \frac{V_\mathrm{max} \cdot C_\mathrm{S}}{K_\mathrm{m} + C_\mathrm{S}}. \tag{3.3}$$

With (3.1), (3.2) and (3.3) search time τ_s can be calculated as:

$$\tau_\mathrm{s} = \left(\frac{V_\mathrm{max}}{v_\mathrm{enz}} - 1\right) \cdot \tau_\mathrm{r} = \frac{K_\mathrm{m} \cdot \tau_\mathrm{r}}{C_\mathrm{S}}. \tag{3.4}$$

A more detailed description of a DEM in biological systems can be found in [7].

3.2 Stochastic behavior

The search and reconfiguration processes are described by a different probability density. In this report, the search process is considered exponentially distributed and the reconfiguration process is assumed to be Γ distributed, which is explained below. This decoupling of distributions in the search and reconfiguration process is one of the most important differences between the discrete event model approach and Gillespie's algorithm. Since an arbitrary distribution can be chosen for the reconfiguration process, essentially any biophysical hypothesis providing the details of the process can be implemented in the system.

3.2.1 Search process

Gillespie's algorithm [3] considers different types of reactions which occur in a volume. In this approach molecules are considered hard spheres. The volume is assumed spatially homogenous where molecules are randomly distributed, in an uniform sense. This distribution does not depend on time. Also, thermal equilibrium is assumed and therefore the collision hazard is independent of time and only depends on the current state of the system. Based on this algorithm the search process is considered exponentially distributed. The mean value of the search time is a function of the buffer content. To draw a sample from an exponential distribution with a changing mean value an inverse calculation method has been used. Sampled search time $\tau_{\mathrm{s,s}}$ has been calculated based on an uniform distribution sample u and a mean value equal to the search time τ_s, as defined in (3.5):

$$\tau_{\mathrm{s,s}} = -ln(1-u) \cdot \tau_\mathrm{s} \tag{3.5}$$

3.2.2 Reconfiguration process

It is not clear what distribution expresses the reconfiguration process. Some experiments show that it is not exponentially distributed, while Gillespie's algorithm assumes an exponential distribution.When the substrate molecule collides with the enzyme, two processes take place. First, the substrate molecule is 'grabbed' by the enzyme, i.e. the bonds are established and both the substrate and enzyme change somewhat in their shape. Second, the enzyme catalyzes a change in conformation of the substrate molecule and then releases the molecule as a product molecule. These processes together are labeled as the reconfiguration process. In this report it is assumed that reconfiguration time τ_{r} is Γ distributed. Coefficient of variation (CV) for the Γ distribution has been chosen to a value of 3.0.

3.3 Interrupting a process

Interrupting the search or reconfiguration process is possible in a substrate-enzyme model. The search process can be interrupted when the substrate concentration changes, in order to recalculate the new search time, or it can be interrupted depending on feedback, presented in Chapter 4. The reconfiguration process can be interrupted if the enzyme receives feedback that affects the reconfiguration rate, presented in Chapter 5. Both processes are described by a different probability density and therefore we first present interruption of an arbitrary probability density. Next, interrupting the exponentially distributed search and Γ distributed reconfiguration process are described.

3.3.1 Arbitrary probability density process interruption

Assume that the time to the next event in the process is described by a probability density $p_1(t)$. At time $t = a$, the probability density describing the process changes to a new density $p_2(t)$. The probability that random variable $T > a$ is defined by z:

$$z = \int_a^\infty p_1(t)dt. \tag{3.6}$$

Number b is defined so that it describes the time in the new distribution that leads to the same cumulative probability as $t = a$ does for the old distribution:

$$\int_b^\infty p_2(t)dt = z. \tag{3.7}$$

With $\tau = t - a$, the conditional probability is therefore given as:

$$Prob\{T \in [\tau, \tau + \delta]|\tau > 0\} = \int_{\tau+b}^{\tau+b+\delta} p_2(s)ds \tag{3.8}$$

The probability density for the new time to finish becomes

$$\tilde{p}_2(\tau|\tau > 0) = \frac{1}{z} p_2(\tau + b) \tag{3.9}$$

Notice $\int_a^\infty p_2(\tau|\tau > 0)d\tau = 1$ which makes this the density that we can use to pull a distribution for the end of the process.

3.3.2 Search process interruption

The search process is considered exponentially distributed (Markovian). If the substrate concentration changes at time $t = a$, probability z that the random variable $T > a$ is defined by:

$$z = \int_a^\infty \lambda_1 e^{-\lambda_1 t} dt = -e^{-\lambda_1 t}\Big|_a^\infty = e^{-\lambda_1 a}. \tag{3.10}$$

Time b describing the time in the new distribution that leads to the same cumulative probability z is defined by:

$$z = \int_b^\infty \lambda_2 e^{-\lambda_2 t} dt = -e^{-\lambda_2 t}\Big|_b^\infty = e^{-\lambda_2 b}, \tag{3.11}$$

$$b = \frac{\lambda_1 a}{\lambda_2}. \tag{3.12}$$

The probability density for the new time to finish then becomes:

$$\tilde{p}_2(\tau|\tau > 0) = \frac{1}{z} p_2(\tau + b) = \frac{\lambda_2 e^{-\lambda_2(\tau+b)}}{e^{-\lambda_2 b}} = \lambda_2 e^{-\lambda_2 \tau} \tag{3.13}$$

Hence (and this is true only for Markov processes, i.e. exponential distributions), once the enzyme is free, a sample is taken from the exponential distribution depending on the substrate concentration. If the next event is a change in substrate concentration during the search process, a new sample is taken from the new exponential distribution and this new sample time is added to the time that has already passed.

3.3.3 Reconfiguration process interruption

In some cases the reconfiguration process is interrupted while reconfiguring a molecule. Reconfiguration time τ_r is assumed to be Γ distributed. Since a Γ distribution is non-Markovian, i.e. it has got a memory, a simple result as with the search process can not be derived. The remaining searchtime after interruption can be calculated with (3.7)-(3.9).

3.4 Simulation results

Before modeling the DEM some parameters have to be converted. One molecule in the DEM corresponds to 1 μMol. Another issue is that the model represents a volume, and buffers work with a list of products. To account for the three-dimensional space a new molecule is placed into the queue at a position drawn from an uniform distribution. With search time, reconfiguration time and initial concentrations known a discrete event model of the substrate-enzyme reaction can be modeled. Figure 3.2 shows the simulation results of the DEM of the substrate enzyme reaction. The complete model and a description of the processes are presented in Appendix E.1. It can be seen that the deterministic results are similar to the ODE results in Figure 2.2. The stochastic results fluctuate around the deterministic results as expected.

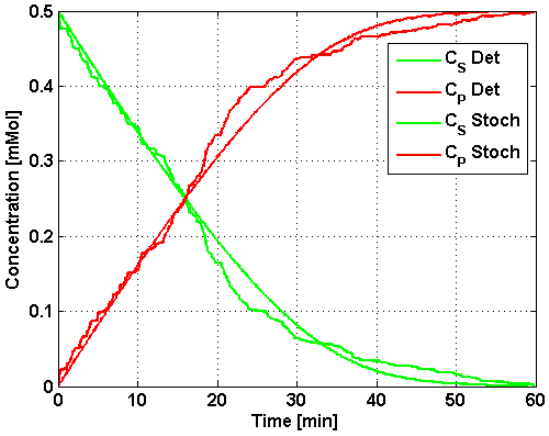

Figure 3.2: Simulation results of the DEM.

Chapter 4

Feedback: Search process

Many substrate-enzyme reactions contain feedback loops that depend on the concentration of the waiting substrate or the concentration of the created product. The biology involved is the following: an enzyme has catalytic binding sites for a substrate molecule, and in addition, it could have a regulatory binding site for a feedback molecule (this can be a substrate or product molecule and will be called feedback molecule from now on). If a feedback molecule binds, it inhibits or activates the enzymes catalysation. Such feedback binding will occur with a typical stochastic time length. In this chapter we introduce activating or inhibiting feedback by respectively accelerating or inhibiting the search process of the enzyme. Activation or inhibition by influencing the reconfiguration process is presented in Chapter 5.

4.1 Inhibiting feedback

When a feedback molecule binds at the regulatory site of the enzyme, this can have an inhibiting effect on the search process. Figure 4.1 presents a graphical representation of downstream inhibiting feedback. When an inhibiting molecule binds, the enzyme changes its structure and prevents substrate molecules to bind at the reconfiguration site until the feedback molecule unbinds. If the enzyme is already reconfiguring a molecule and a feedback molecule binds, the molecule under reconfiguration is not affected. After the product molecule unbinds from the enzyme, search for new substrate molecules is inhibited.

The model used to simulate the influence of feedback is an enzyme with maximal activity $V_{\max} = 100.0$, Michaelis-Menten constant $K_{\mathrm{m}} = 40.0$ and reconfiguration time $\tau_{\mathrm{r}} = 0.01$. It is assumed that the search time between molecules at the regulatory and catalytic sites are similar and the binding time at the regulatory site is assumed to be 1 second (0.0167 min) and is denoted byτ_{fb} . Results of simulations with a single enzyme receiving inhibiting feedback from the substrate molecules are presented in Figure 4.2. This figure presents the reaction rate depending on the substrate concentration. Results with deterministic settings are presented in green, stochastic settings are presented in blue and

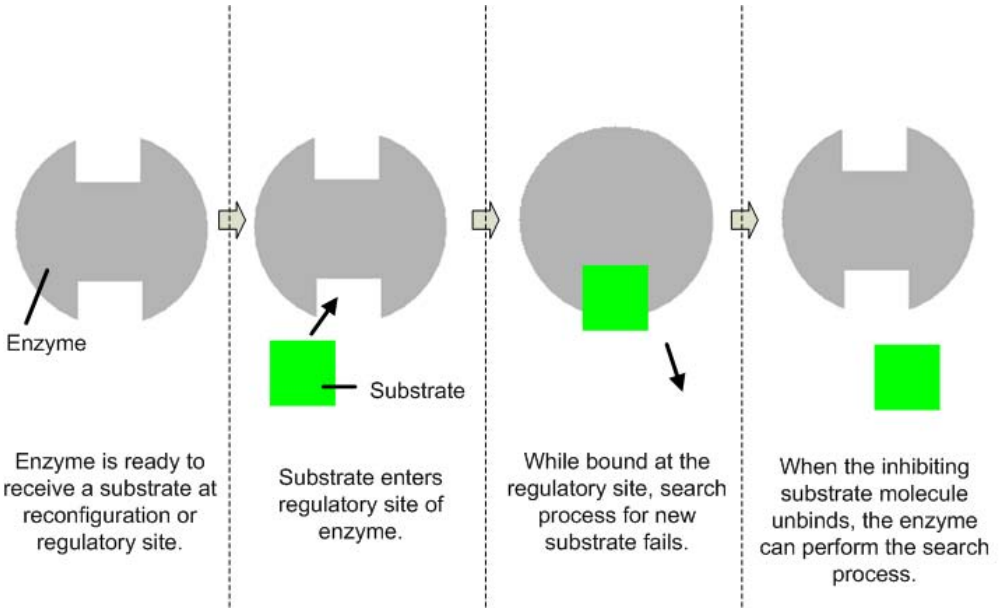

Figure 4.1: Graphical representation of inhibiting search time feedback from substrate S.

the dotted line represents the enzymic rate without feedback. It can be seen that both the stochastic and deterministic system with inhibiting feedback result in a lower activity. However, the deterministic model gives a much higher activity than the stochastic system. Since the search processes are similar, both molecules bind at the enzyme when the search processes start at the same time in the deterministic model. In the stochastic model, the chance of binding a feedback molecule first is 50%. If a feedback molecule binds first, the search process at the reconfiguration site is inhibited and a new search process starts when the feedback molecule unbinds. This explains that the stochastic curve is about half of the deterministic curve.

The effect of varying the time a feedback molecule is bound to the enzyme is presented in Figure 4.4. Results of the deterministic system are presented by the blue line and results of the stochastic system are presented in red. It is expected that for a binding time of zero the activity is maximal and decreases with increasing binding time, as shown by the stochastic results. The deterministic results exist of two different regions, $\tau_{fb} < \tau_r$ and $\tau_{fb} \geq \tau_r$. Figure 4.3 shows sample paths of a setting in these regions, along with the case $\tau_{fb} = \tau_r$. The arrows represent search times, the red blocks the binding time of a molecule at the catalytic site and the grey striped blocks the binding time at the regulatory site. If $\tau_{fb} < \tau_r$, enumerated by 1, the reaction rate equals the time two feedback molecules bind and unbind:

$$V = \frac{1}{2 \cdot \tau_s + 2 \cdot \tau_{fb}} \quad \text{if } \tau_{fb} < \tau_r, \tag{4.1}$$

For $\tau_{fb} = \tau_r$, enumerated by 2, the reconfiguration rate is not affected by the feedback molecules and if $\tau_{fb} \geq \tau_r$, enumerated by 3, the reaction rate equals the binding rate of one feedback molecule:

$$V = \frac{1}{\tau_s + \tau_{fb}} \quad \text{if } \tau_{fb} \geq \tau_r, \tag{4.2}$$

To eliminate these effects and to be able to make a fair comparison between stochastic and deterministic models only the stochastic feedback models are used in the rest of this chapter.

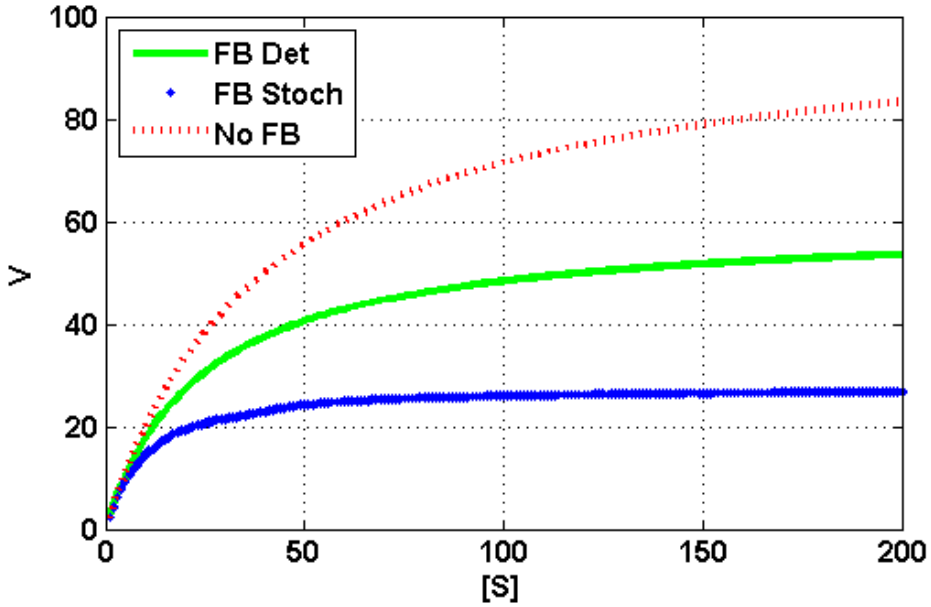

Figure 4.2: Inhibiting search process feedback simulation results.

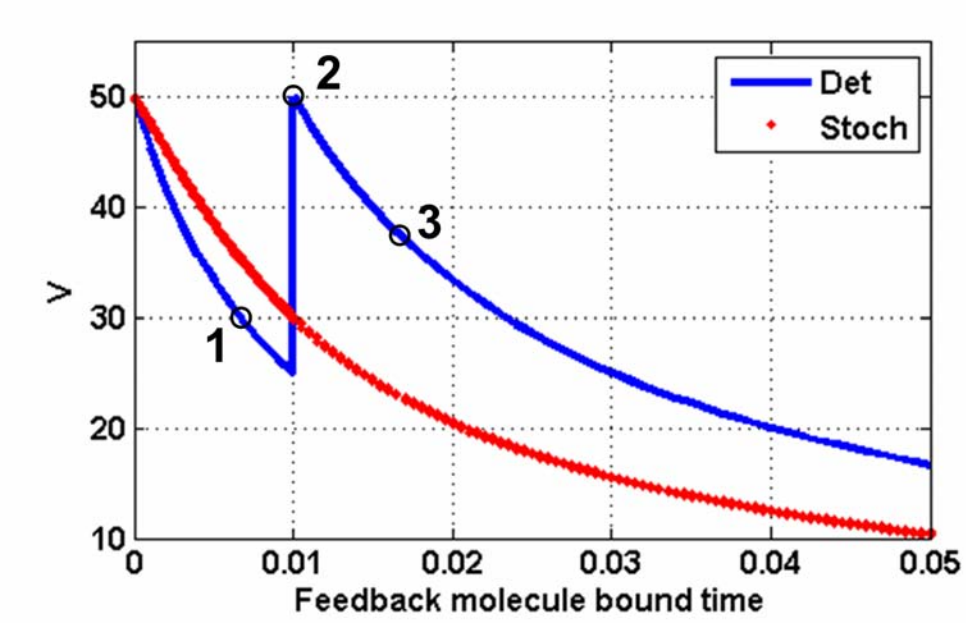

Figure 4.3: Sample paths of 3 cases with different feedback molecule bound times. Number of substrate products is 40.

It is possible that the search time for a substrate molecule and the catalytic site is not equal to the search time for a substrate molecule and the regulatory site. This can be caused by the positions of the sites on the enzyme or difference in attraction force of the sites. For different search times of the feedback molecules, different shapes of specific activity can be modeled. If the influence of the feedback is increased (feedback

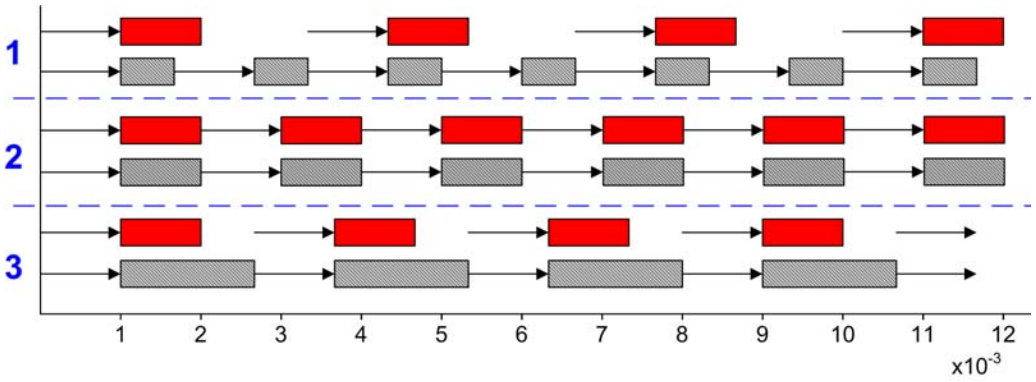

Figure 4.4: Sample paths of 3 cases with different feedback molecule bound times. Number of substrate products is 40.

molecule binding time of 0.2), and feedback search times are much longer ($K_{\mathrm{m}} = 20000$ and $\tau_{\mathrm{r}} = 0.001$), specific activity decreases for increasing substrate concentration from a certain point, see Figure 4.5. Each point has been simulated 5 times, presented by the blue dots in the upper figure. The red line shows the mean of these simulations and the lower figure the coefficient of variation of the points.

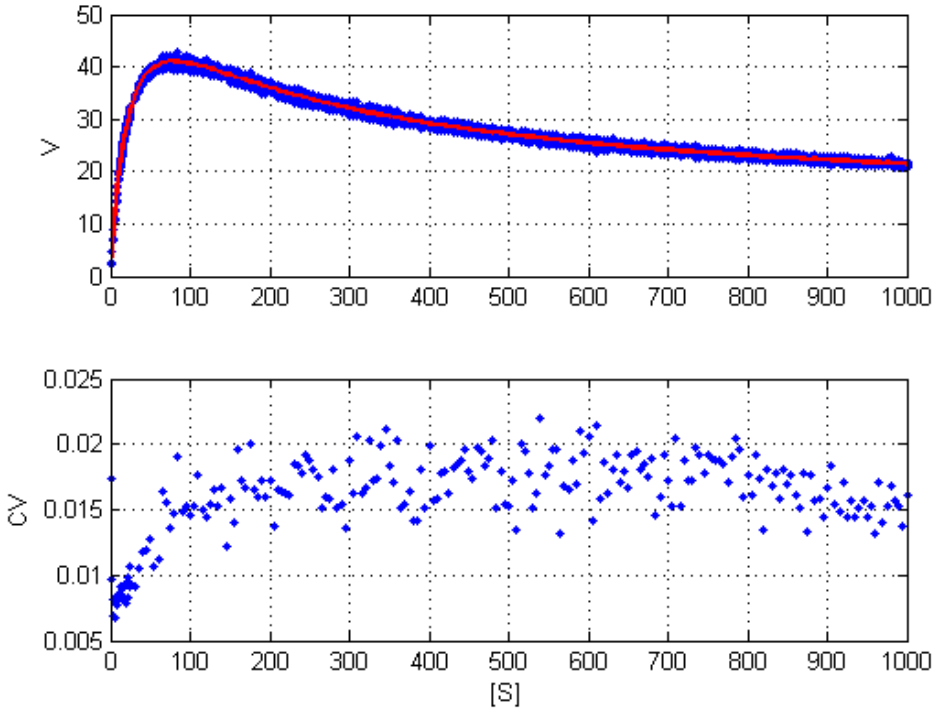

Figure 4.5: Inhibiting search process feedback simulation results.

4.2 Activating feedback

In contrast to inhibiting feedback presented in Section 4.1, activating feedback accelerates the search process. The search process with activating feedback is assumed to be ten to a hundred times faster than a normal search process. A graphical representation of activating feedback is presented in Figure 4.6. Results of specific activity V versus substrate concentration $[S]$ are presented in Figure 4.7. The black dotted line presents the specific activity without feedback, i.e. the Michaelis-Menten curve. The red and green lines present the average values from the stochastic simulation results which increases the search process respectively ten and a hundred times if a feedback molecule is bound. It can be seen that the specific activity is much higher for activating feedback, as expected, while the difference between an increase of search time by 10x or 100x does not make a big difference.

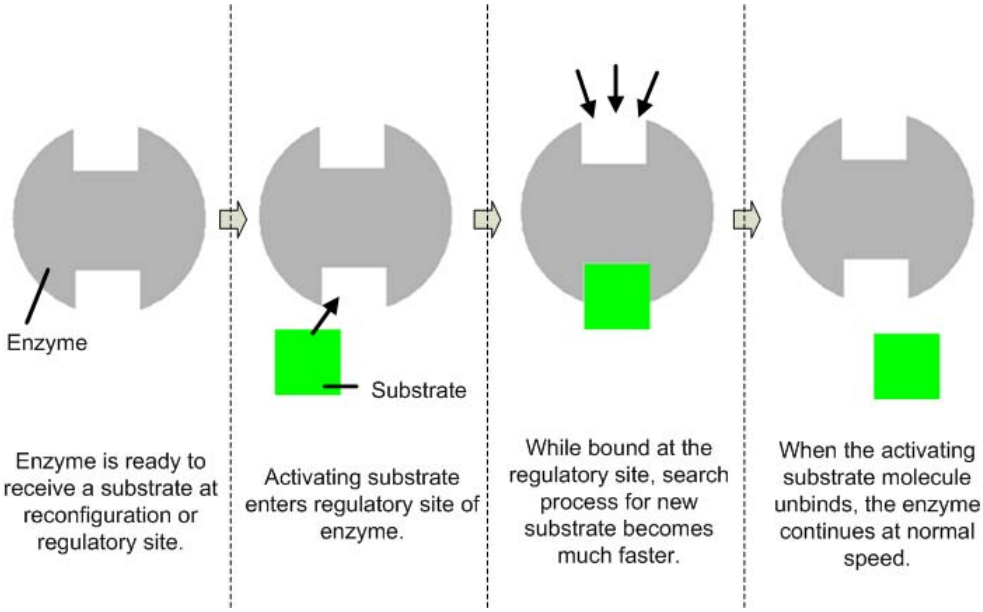

Figure 4.6: Graphical representation of activating search time feedback from substrate S.

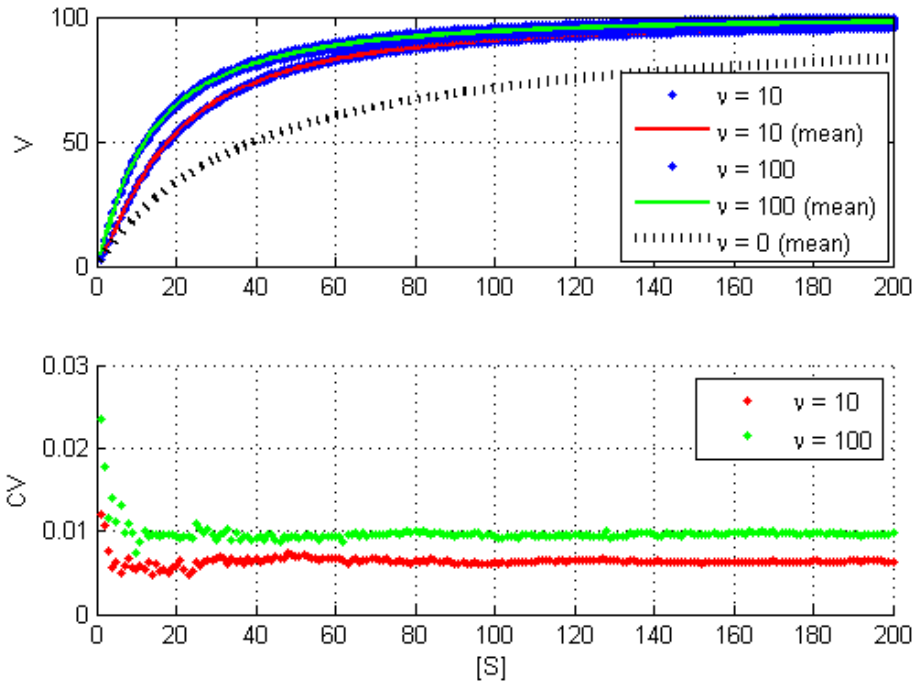

Figure 4.7: Activating search process feedback simulation results.

4.3 Inhibiting and activating feedback

In some cases in a pathway consisting of substrate-enzyme reactions, some enzymes are are inhibited by downstream energy molecules and activated by upstream energy molecules, see Chapter 7. The enzymes reconfigure a substrate molecule together with an energy molecule into a product molecule and an energy molecule with lower energy. These energy-molecules can also bind with the enzyme at the regulatory site and activate or inhibit the reaction. A discrete event model representation of this situation is presented in Figure 4.8. Feedback is indicated with ν in the model, and there is downstream inhibition and upstream activation.

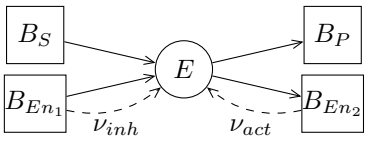

Figure 4.8: DEM representation of a process with inhibiting and activating feedback.

Results of stochastic simulations with this model are presented in Figure 4.9. The blue lines represent the substrate concentration and the red lines represent the product concentration. Also results without feedback are presented (dotted lines). Parameters of the feedback search time are similar to the reconfiguration substrate search time and the influence of an activating feedback is a 50 times faster search process. The energy molecules concentrations are similar to the substrate and product concentrations. It can be seen that in the first part the conversion rate is slower than without feedback, and after this point the rate is faster than for the corresponding concentrations in the results without feedback. These results are also expected and this sigmoidal shape is well-known in biology. Substrate energy concentration is high at the start and therefore a lot of inhibiting feedback molecules bind at the regulatory site of the enzyme. While the substrate concentration decreases and product concentration increases, less inhibiting and more activating feedback molecules bind with the enzyme resulting in the fast rate at the end.

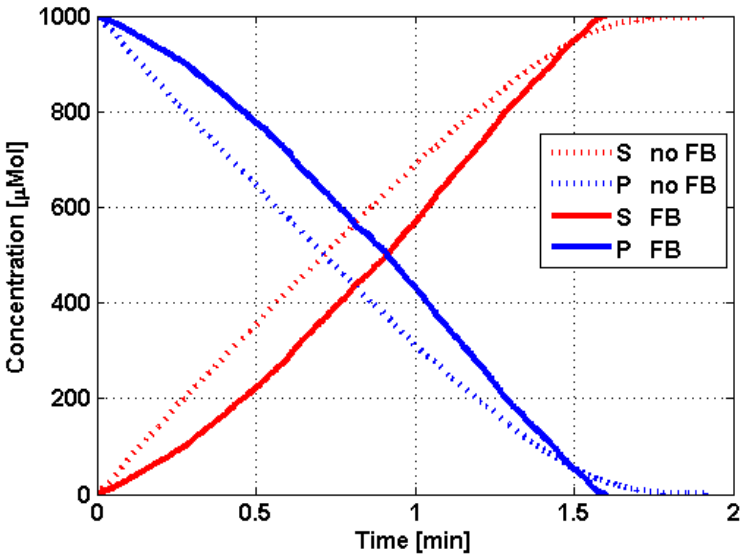

Figure 4.9: Activating and inhibiting search process feedback simulation results.

4.4 Combining enzymes

In some processes, like the destruction of Adenosine monophosphate (AMP)[1], specific activity decreases at a certain concentration and increases for a larger concentration, see Figure 4.10. In this case AMP is degraded by AMP deaminase and AMP phosphatase. AMP deaminase has a strong positive cooperativity for AMP and seems to be almost inactive at normal AMP concentrations, what explains the left part of the figure. For higher concentrations, AMP acts as an effector for AMP phosphatase resulting in an increase of degradation for higher concentrations of AMP.

This situation can be modeled by combining an enzyme with downstream inhibiting feedback and a normal enzyme, see Figure 4.11. Both enzymes are competing for the substrate molecules. Results are presented in Figure 4.12 and the curve is similar to the curve in Figure 4.10. Another option is to combine the enzyme with feedback with an enzyme that becomes active at a certain concentration level. Results of this system are presented in Figure 4.13, the enzyme becomes active when the substrate concentration ≥ 500.

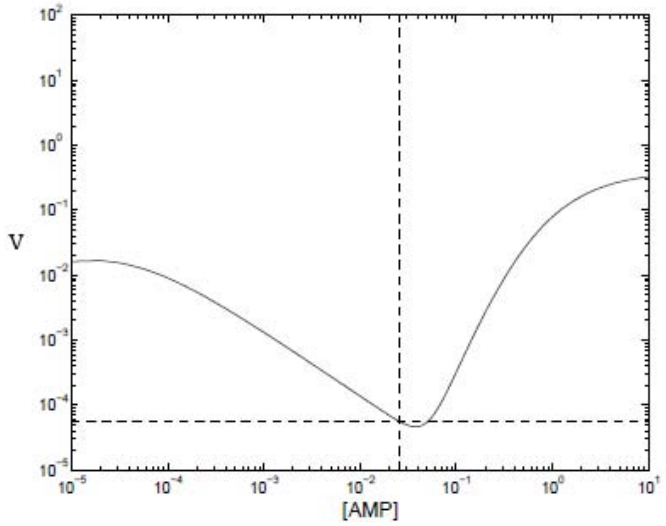

Figure 4.10: AMP destruction function.

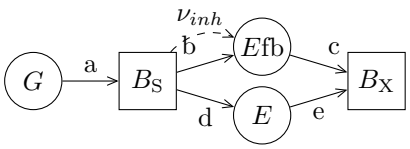

Figure 4.11: DEM representation of a combination of a feedback and normal enzyme.

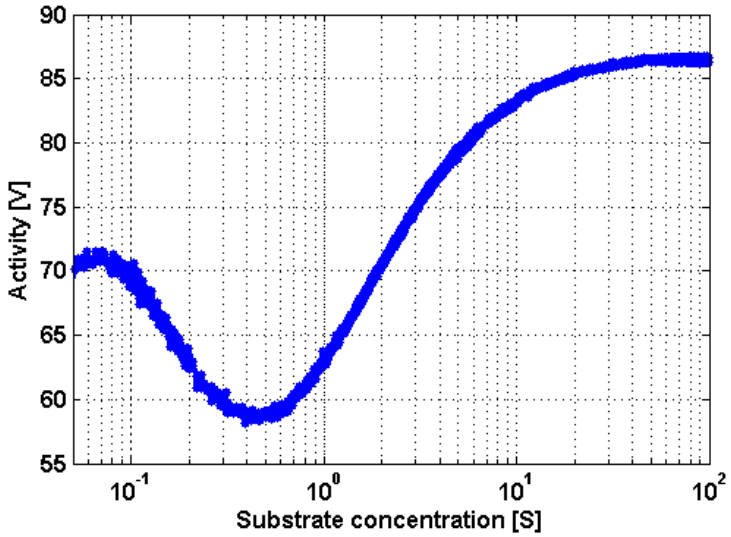

Figure 4.12: Combination of feedback and normal enzyme simulation results.

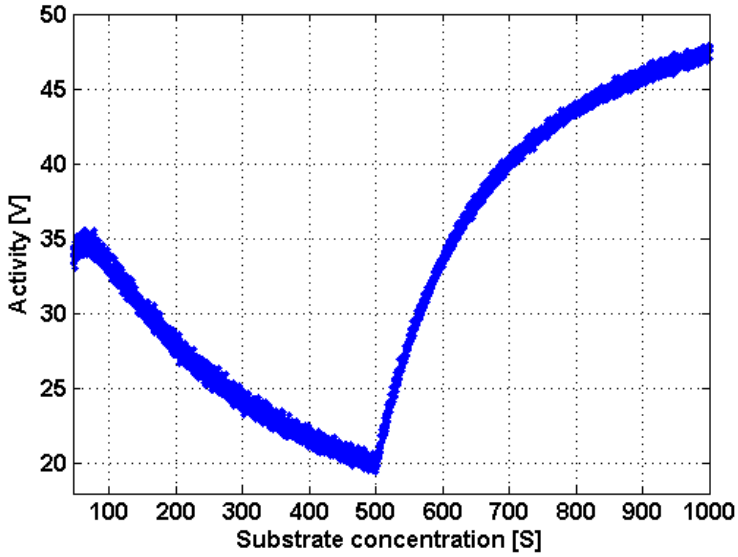

Figure 4.13: Combination of feedback and normal enzyme simulation results, normal enzyme is activated when [S] = 500.

4.5 Conclusion

In this chapter, inhibiting and/or activating feedback has been implemented in the DEM of an enzyme-substrate reaction. The activation or inhibition occurs in the search process. If an inhibiting feedback molecule binds with the enzyme, the enzyme can not receive a new substrate molecule and therefore the search process is inhibited until the molecule unbinds. If an activating feedback molecule binds, search process becomes 10-100 times faster. When the feedback parameters are changed a parabolic shape in the result of the specific activity versus substrate concentration can be created. Also results of coupling this enzyme with a 'normal' enzyme are presented. Only downstream inhibition and upstream activation are discussed, downstream activation and upstream inhibition can be modeled and simulated similarly.

Chapter 5

Feedback: Reconfiguration process

Instead of inhibition or activation by failing or accelerating the search process presented in Chapter 4, inhibition or activation can also occur by deceleration or acceleration of the reconfiguration process. This chapter presents feedback of substrate or product molecules that affect the reconfiguration process of the enzyme. In Section 5.1 we implement inhibiting feedback of product molecule P in the DEM of the substrate-enzyme reaction. In Section 5.2 activating feedback of substrate molecule S is modeled and simulated. Section 5.3 presents feedback of both substrate and product molecules.

5.1 Inhibiting feedback

In this section inhibiting feedback of product molecule P is implemented in the substrate-enzyme model from the Chapter 3. A graphical representation of this reaction with feedback is shown in Figure 5.1. If the concentration of product molecules becomes larger, reaction speed slows down because more inhibiting feedback molecules will bind. The binding time of a feedback molecule with the enzyme is chosen to be 0.1 time units. Effect of this feedback is that the reconfiguration rate halves until the feedback molecule unbinds from the regulatory site. Results of introducing inhibiting feedback from product molecules with different parameters for the search time between product molecule and regulatory site of the enzyme are presented in Figure 5.2. The solid lines present the results without feedback and the dotted lines present results with feedback. It can be seen that more feedback slows the reaction down, as expected.

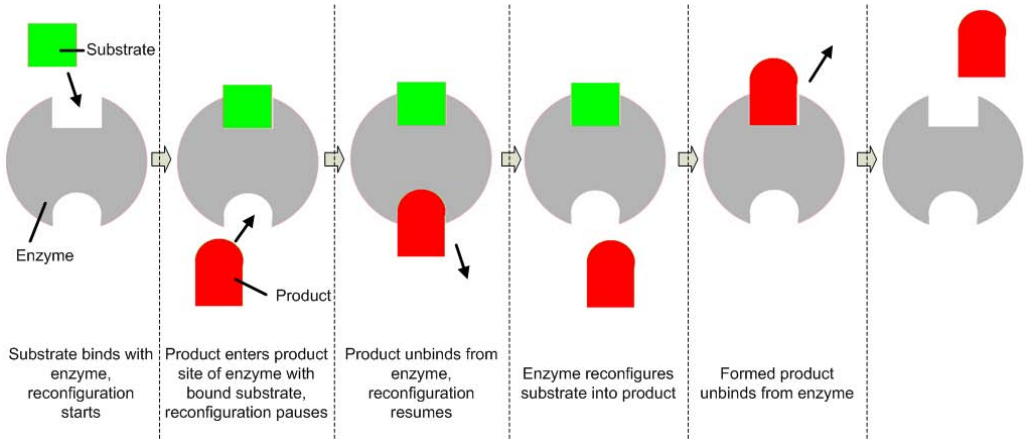

Figure 5.1: Graphical representation of inhibiting feedback from product P.

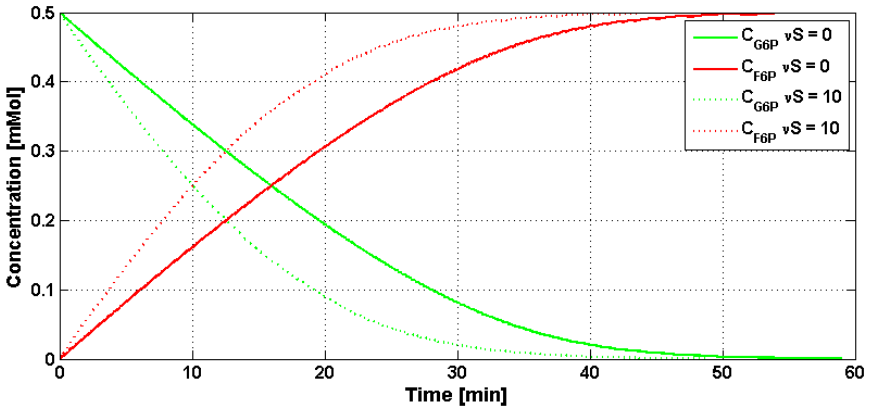

Figure 5.2: Results of simulations with inhibiting feedback from P.

5.2 Activating feedback

The model from Section 5.1 simulated inhibiting upstream feedback to slow down the enzyme depending on the product molecules concentration. In contrast, substrate molecules can give activating feedback to the enzyme depending on the substrate concentration. A graphical representation of this reaction is shown in Figure 5.3. In this model is assumed that if a substrate molecule is bound at the regulatory site of the enzyme, reconfiguration rate doubles until the activating enzyme unbinds. Results of introducing activating feedback from substrate molecules with different parameters for the search time between substrate molecule and regulatory site of the enzyme are presented in Figure 5.4. It can be seen that reaction speed increases with increasing feedback.

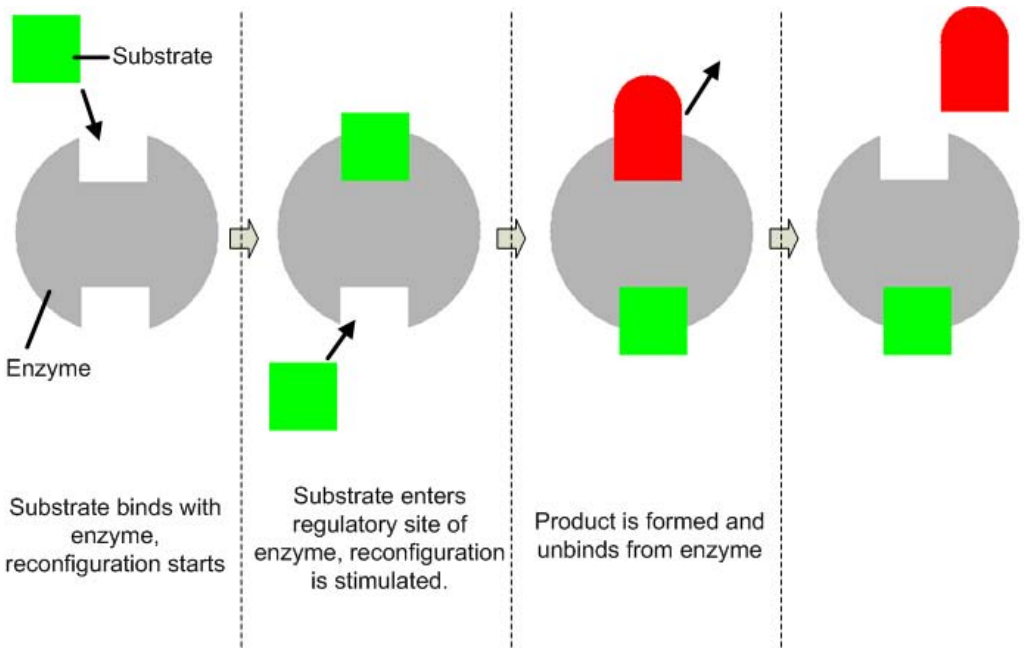

Figure 5.3: Graphical representation of activating feedback from substrate S.

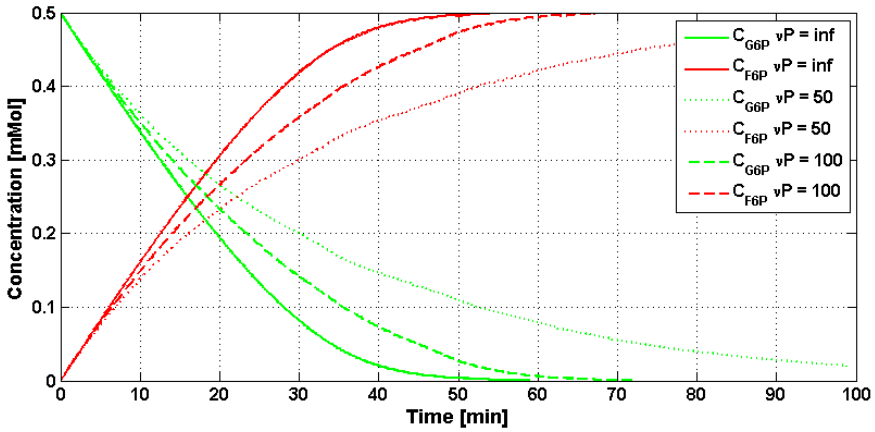

Figure 5.4: Results of simulations with activating feedback from S.

5.3 Inhibiting and activating feedback

Upstream inhibiting feedback (from product P) or downstream activating feedback (from substrate S) largely regulate the reaction speed in reality, for instance in glycolysis. This is a combination of the previous two models. Only one feedback molecule can bind at the enzyme at a time. A graphical representation of this reaction with feedback is shown in Figure 5.5. Results of the model with motivating feedback from substrate molecules with different search time rates is presented in Figure 5.6. The legend shows the type of molecule, ν_S and ν_P. If both parameters are set to zero, the enzyme receives no feedback and the results are similar to the ODE model.

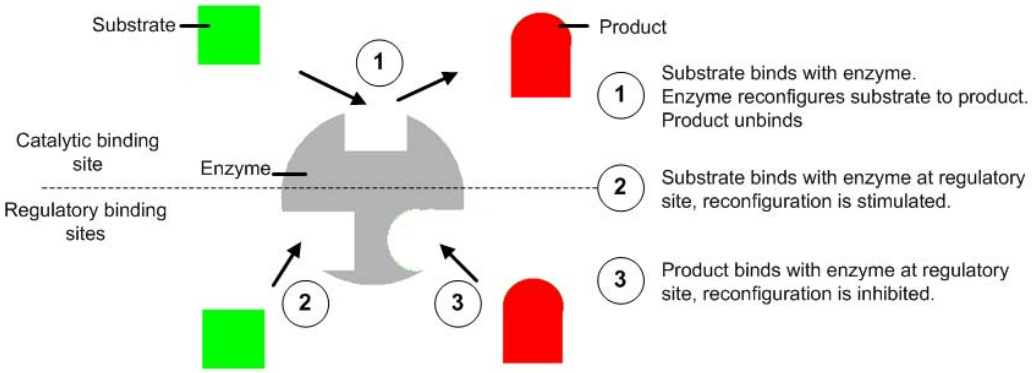

Figure 5.5: Graphical representation of feedback from S and P.

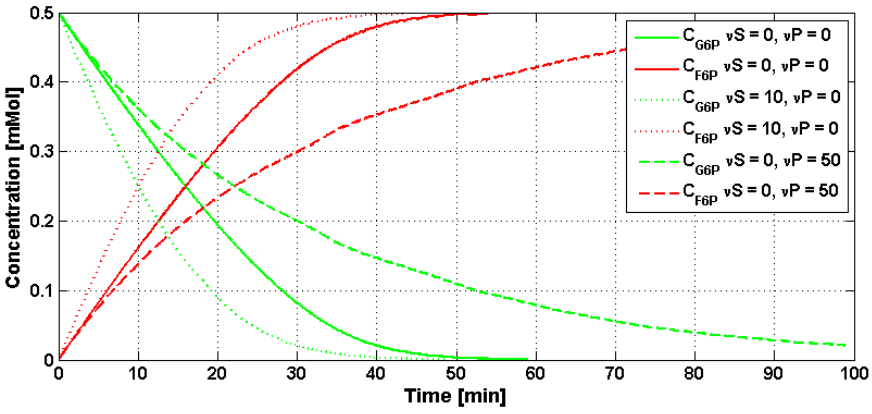

Figure 5.6: Results model with feedback from S and P.

5.4 Conclusion

In this chapter feedback of the substrate or product molecules affecting the reconfiguration process has been introduced in a substrate-enzyme reaction. An activating feedback of substrate molecules and an inhibiting feedback of product molecules has been presented.

Chapter 6

Reversible reaction

A substrate-enzyme reaction can be irreversible or reversible. Reactions described in the previous chapters have all been irreversible, i.e. enzyme E only reconfigures substrate S into product P and not the other way around. In reversible reactions the enzyme also reconfigures product molecule P into substrate molecule S. A reversible reaction scheme is presented in (6.1).

$$S + E \rightleftarrows P + E. \tag{6.1}$$

The reaction in (6.2) is one of the reactions in the glycolysis pathway, see [4]. In this reaction Glucose-6-Phosphate *G6P* is reconfigured (isomerization) into Fructose-6-Phosphate *F6P* by enzyme Phosphoglucoisomerase *Pgi*. This reaction is freely reversible under normal cell conditions. However, it is often driven forward because of a low concentration of *F6P*, which is constantly consumed during the next step of glycolysis. Under conditions of high *F6P* concentration this reaction readily runs in reverse. This phenomenon can be explained with Le Chatelier's Principle.

$$G6P + Pgi \rightleftarrows F6P + Pgi. \tag{6.2}$$

In Section 6.1, the reversible reaction has been modeled and simulated as an ODE model. Section 6.2 presents the reaction modeled and validated as a DEM.

6.1 ODE model

The set of ODEs for this reaction is:

$$\frac{dC_{G6P}}{dt} = -v_{Pgi}, \tag{6.3}$$

$$\frac{dC_{F6P}}{dt} = v_{Pgi}, \tag{6.4}$$

with v_{Pgi} the specific activity of the enzyme. From [4]:

$$v_{Pgi} = \frac{V_{\max} \cdot \left(C_{G6P} - \frac{C_{F6P}}{K_{\mathrm{eq},Pgi}} \right)}{K_{\mathrm{m},G6P} \cdot \left(1 + \frac{C_{F6P}}{K_{\mathrm{m},F6P}} \right) + C_{G6P}}, \tag{6.5}$$

where $V_{\max}$ denotes the maximal reaction rate, C_x the concentration of x, $K_{\mathrm{m,x}}$ denotes the Henri-Michaelis-Menten constant for x and $K_{\mathrm{eq},Pgi}$ is the equilibrium constant. The Pgi parameters are presented in Appendix A, Table A.

Results of simulations with this ODE system with initial C_{G6P} and C_{F6P} amounts set to respectively 1.0 and 0.0 mMol are shown in Figure 6.1. The equilibrium concentrations can be calculated ($v_{Pgi} = 0$), if total concentration T is known, by:

$$C_{G6P} - \frac{C_{F6P}}{K_{\mathrm{eq},Pgi}} = 0, \tag{6.6}$$

$$C_{G6P} + C_{F6P} = T. \tag{6.7}$$

For $F_{G6P}(0) = 1.0$mMol and $C_{F6P}(0) = 0.0$mMol this results in equilibrium concentrations $C_{G6P} = 10/13 = 0.77$mMol and $C_{F6P} = 3/13 = 0.23$mMol.

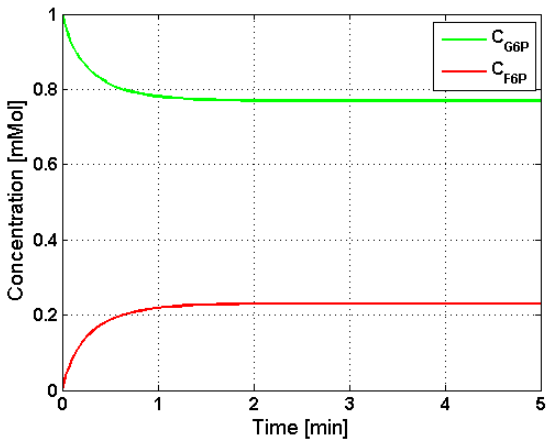

Figure 6.1: ODE results for $C_{G6P}(0) = 1$ and $C_{F6P}(0) = 0$.

6.2 Discrete event model

The reversible reaction can be modeled as a discrete event model, with a similar approach as the substrate-enzyme and feedback DEMs. A DEM representation of the reversible reaction is presented in Figure 6.2. The blocks B_S and B_P represent the buffers. The buffers contain the concentration of respectively substrate (*G6P*) and product (*F6P*) molecules . The reaction process (enzyme) is presented by R_E. Two different approaches have been used for modeling the deterministic and stochastic discrete event model of the reversible reaction. The deterministic approach is described in Section 6.2.1 and the stochastic approach in Section 6.2.2.

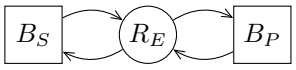

Figure 6.2: DEM representation of a reversible reaction.

6.2.1 Deterministic approach

The deterministic DEM of the reversible reaction represents the overall activity of the molecules. When overall activity v_{Pgi} equals zero, an equilibrium has been reached and the model will not reconfigure any substrate or product molecules. Actually, in equilibrium, the forward and backward reaction rate are constant. For the deterministic model it is computationally less expensive to simulate the overall rate instead of both the forward and backward rates, yielding the same results. Overall reaction time Δt (sum of search and reconfiguration times) of one molecule is denoted by the absolute value of the specific activity of enzyme $|v_{Pgi}|$ multiplied by the number of enzymes. If $v_{Pgi} > 0$ more *G6P* molecules are reconfigured into *F6P* molecules and if $v_{Pgi} < 0$ more *F6P* molecules are reconfigured into *G6P* molecules. Total reaction time of one molecule is:

$$\Delta \mathrm{t} = \tau_\mathrm{s} + \tau_\mathrm{r} = \frac{1}{|v_{Pgi}| \cdot V}. \tag{6.8}$$

Results of simulations with the deterministic DEM for $C_{G6P}(0) = 1$ and $C_{F6P}(0) =$ 0mMol are presented in Figure 6.3. A comparison between the DEM and ODE simulation results is shown in Figure 6.4. It can be seen that the DEM results track the ODE results perfectly with a stepsize of 1μMol.

6.2.2 Stochastic approach

For stochastic simulations the deterministic approach, using the overall rate, is insufficient. For this approach with stochastic processes, the concentrations of the substrate and product molecules would fluctuate very little near an equilibrium, since the reaction speed v_{Pgi} is almost zero. In real life, the concentrations fluctuate much more due to the stochastic fluctuations in the forward and backward reaction rates. Therefore, the stochastic DEM of the reversible reaction describes both the forward and backward reaction. Forward reaction speed v_F is assumed to be:

$$v_\mathrm{F} = \frac{V_\mathrm{max} \cdot C_{G6P}}{K_{\mathrm{m},G6P} \cdot \left(1 + \frac{C_{F6P}}{K_{\mathrm{m},F6P}}\right)}. \tag{6.9}$$

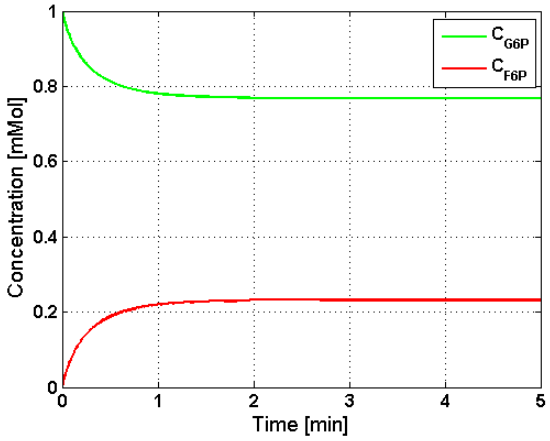

Figure 6.3: Deterministic DEM results for $C_{G6P}(0) = 1$ and $C_{F6P}(0) = 0$.

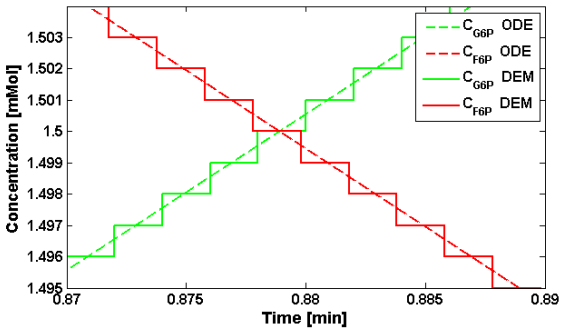

Figure 6.4: Comparison between ODE and DEM results for $CG6P(0) = 1$ and $C_{F6P}(0) = 2$.

Overall reaction speed v_{Pgi} is the forward reaction speed extracted by the backward reaction speed v_{B}:

$$v_{Pgi} = v_{\mathrm{f}} - v_{\mathrm{b}}, \tag{6.10}$$

therefore:

$$v_{\mathrm{B}} = \frac{V_{\max} \cdot \frac{C_{F6P}}{k_{\mathrm{eq},Pgi}}}{K_{\mathrm{m},G6P} \cdot \left(1 + \frac{C_{F6P}}{K_{\mathrm{m},F6P}}\right)}. \tag{6.11}$$

Simulation results of the stochastic DEM are presented in Figure 6.5. It can be seen that after the startup-phase the concentrations fluctuate around the equilibrium levels.

6.3 Conclusion

In this chapter a reversible substrate-enzyme reaction has been modeled as a discrete event model. The deterministic and stochastic models are constructed with a different approach,

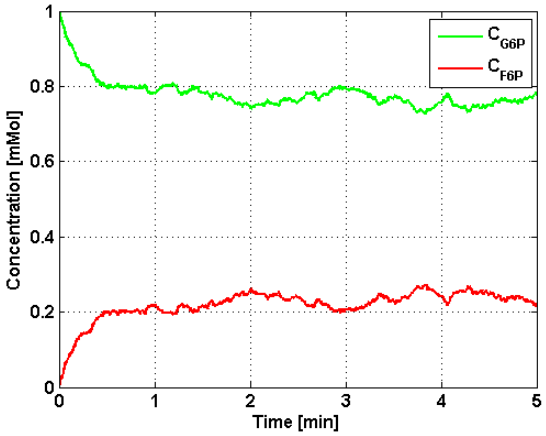

Figure 6.5: Stochastic DEM results for $C_{G6P}(0) = 1$ and $C_{F6P}(0) = 0$.

resulting in less calculations for the deterministic model. Results from simulations with the deterministic model are verified with results of simulations of a set of ODEs.

Chapter 7

EMP Pathway

With discrete event models for a substrate enzyme reaction, reactions with inhibiting and/or activating feedback and reversible reactions a biological network or pathway can be modeled. As a test-case the first steps of glycolysis are modeled and analyzed in this chapter. Glycolysis is the metabolic pathway that converts glucose $C_6H_{12}O_6$, into pyruvate $C_3H_3O_3^-$ while releasing energy. Glycolysis is thought to be the archetype of a universal metabolic pathway. It occurs, with variations, in nearly all organisms. The wide occurrence of glycolysis indicates that it is one of the most ancient known metabolic pathways.
The most common type of glycolysis is the Embden-Meyerhof-Parnas (EMP) pathway, which was first discovered by Gustav Embden, Otto Meyerhof and Jakub Karol Parnas in 1918. This pathway is a sequence of twelve reactions. We consider the first steps, see Figure 7.1. The first step in glycolysis is phosphorylation of glucose *Gluc* by enzymes called Hexokinase *Hk* to form glucose-6-phosphate *G6P*. This reaction consumes high energy compound adenosine triphosphate *ATP* and a product is adenosine diphosphate *ADP*. *G6P* is then rearranged into fructose-6-phosphate *F6P* by enzyme phosphoglucoisomerase *Pgi*. This reaction is freely reversible under normal cell conditions. Enzyme *Pfk* works similar as enzyme *Hk*. It removes a phosphate from *ATP* and binds it to *F6P*, generating fructose-1,6-bisphosphate *F1,6bP* and *ADP*.

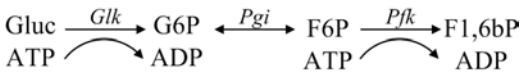

Figure 7.1: First steps EMP-pathway.

Section 7.1 presents the ODE and DEM models of the EMP-pathway and results of simulations are compared. In Section 7.2, the model is extended with activating feedback of *ADP* molecules and inhibiting feedback of *ADP* molecules. Section 7.3 presents the stochastic behavior of the model. The transient model is modified to a steady-state model in Section 7.4 and results are analysed in Section 7.5.

7.1 ODE and DEM models

The enzymic reactions from the first steps of the EMP-pathway shown in Figure 7.1 are decoupled in (7.1).

$$Gluc + ATP \xrightarrow{v_{Hk}} G6P + ADP, \tag{7.1a}$$

$$G6P \overset{v_{Pgi}}{\leftrightarrow} F6P, \tag{7.1b}$$

$$F6P + ATP \xrightarrow{v_{Pfk}} F1,6bP + ADP. \tag{7.1c}$$

Kinetic rate equations are shown in (7.2) and taken from [4]. Reaction (7.1b) is a reversible reaction but can also be considered irreversible. Both kinetic rate equations (7.2b and 7.2c) are used in this chapter. The parameters used are presented in Appendix A, Table A.3.

$$v_{Hk} = \frac{V_{\max} \cdot C_{Gluc} \cdot C_{ATP}}{\left(K_{\mathrm{m},Gluc} + C_{Gluc}\right) \cdot \left(K_{\mathrm{m},ATP} + C_{ATP}\right)}, \tag{7.2a}$$

$$v_{Pgi} = \frac{V_{\max} C_{\mathrm{G6P}}}{K_{\mathrm{m,G6P}} + C_{\mathrm{G6P}}}, \tag{7.2b}$$

$$v_{Pgi,\mathrm{rev}} = \frac{V_{\max} \cdot \left(C_{\mathrm{G6P}} - \frac{C_{F6P}}{K_{\mathrm{eq},Pgi}}\right)}{K_{\mathrm{m,G6P}} \cdot \left(1 + \frac{C_{F6P}}{K_{\mathrm{m},F6P}}\right) + C_{\mathrm{G6P}}}, \tag{7.2c}$$

$$v_{Pfk} = \frac{V_{\max} \cdot C_{F6P}^{n} \cdot C_{ATP}}{\left(K_{\mathrm{m},F6P}^{n} + C_{F6P}^{n}\right) \cdot \left(K_{\mathrm{m},ATP} + C_{ATP}\right)}. \tag{7.2d}$$

According to (7.1) and (7.2) the ordinary differential equations describing the change in concentration in the system can be expressed as:

$$\frac{dC_{Gluc}}{dt} = -v_{Hk} + v_{\mathrm{arr}}, \tag{7.3a}$$

$$\frac{dC_{\mathrm{G6P}}}{dt} = v_{Hk} - v_{Pgi}, \tag{7.3b}$$

$$\frac{dC_{F6P}}{dt} = v_{Pgi} - v_{Pfk}, \tag{7.3c}$$

$$\frac{dC_{F1,6bP}}{dt} = v_{Pfk} - v_{\mathrm{exit}}, \tag{7.3d}$$

$$\frac{dC_{ATP}}{dt} = -v_{Hk} - v_{Pfk} + v_{\mathrm{conv}}, \tag{7.3e}$$

$$\frac{dC_{ADP}}{dt} = v_{Hk} + v_{Pfk} - v_{\mathrm{conv}}. \tag{7.3f}$$

Results of the DEM and ODEs for these first steps in the EMP-pathway without reversible *Pgi* reaction are presented in Figure 7.2. Initial concentrations are $C_{Gluc}(0) = 1.0$mMol, $C_{ATP}(0) = 2.0$mMol, enzyme concentrations are 0.05 μmol and all other initial concentrations are zero. The ODE results are presented by dotted lines and the DEM results are presented by solid lines. It can be seen that the ODE and DEM results are similar. In Figure 7.3, the first steps of the EMP-pathway has been simulated with the reversible reaction (*G6P* molecules are reconfigured into *F6P* molecules and the other way around). The ODEs and DEM results are similar. A comparison between the results with and without reversibility is presented in Figure 7.4. With the reversible reaction

$G6P$ concentration is higher and $F6P$ concentration is lower than without reversibility, as expected.

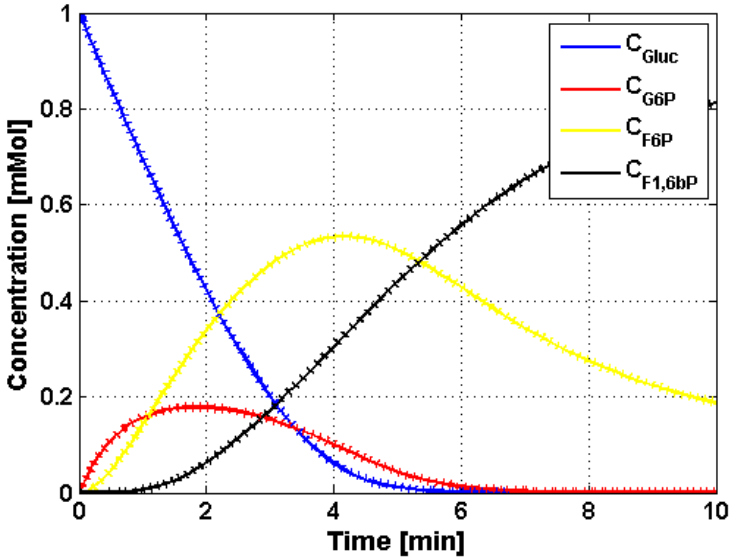

Figure 7.2: ODE (dotted line) and DEM (line) results without reversible reaction and $C_{Gluc}(0) = 1.0$ and $C_{ATP}(0) = 2.0$mMol.

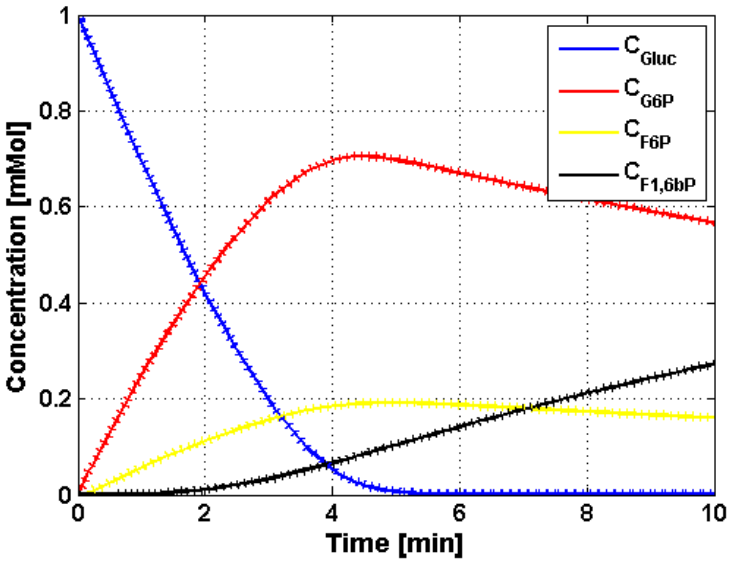

Figure 7.3: ODE (dotted line) and DEM (line) results with reversible reaction and $C_{Gluc}(0) = 1.0$ and $C_{ATP}(0) = 2.0$mMol.

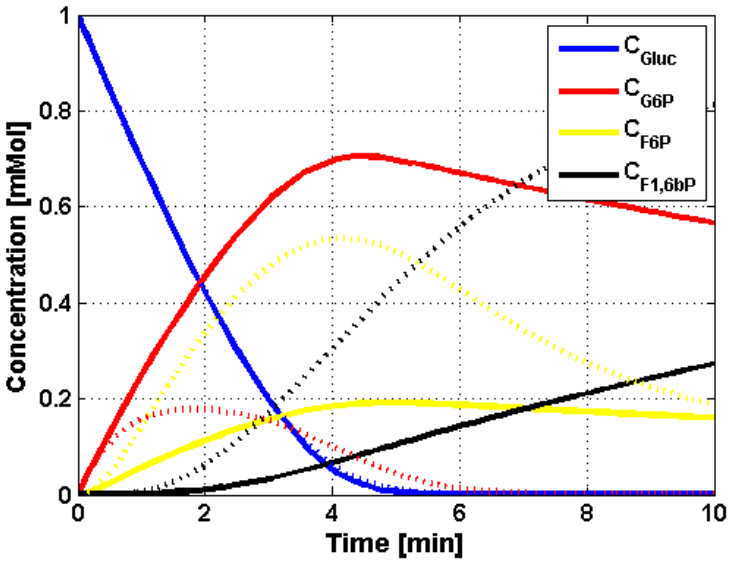

Figure 7.4: Results without (line) and with (dotted line) reversibility and $C_{Gluc}(0) = 1.0$ and $C_{ATP}(0) = 2.0$mMol.

7.2 Including feedback

In Section 7.1 the early steps of glycolysis have been modeled as a DEM. In this model a number of simplifications that are not biologically justifiable have been made. Among these, no downstream feedback inhibition or upstream activation on any enzyme in the pathway has been assumed. In reality, however, the rate of glycolysis is considered largely regulated by such inhibition and activation. The enzyme phosphofructokinase *Pfk*, for example, is widely believed to be the key regulator of glycolysis (and therefore much of glucose metabolism) in bacteria, yeast and many other organisms. This enzyme catalyzes the phosphorylation of *F6P* to *F1,6bP*, a process that is *ATP*-dependent. In addition, in both bacteria and yeast, *Pfk* is inhibited by *ATP*. In particular, *Pfk* has two *ATP*-binding domains, one catalytic and one regulatory. When *ATP* is bound at this regulatory site, *Pfk* takes on a conformation with a relatively low *ATP* affinity at the catalytic site. In contrast, *ADP*, the hydrolyzed form of *ATP*, activates *Pfk*. That is, when bound to *Pfk*'s regulatory site, *ADP* increases the enzyme's catalytic affinity for *ATP*. These mechanisms are widely believed to provide the main control of the rate of glycolysis.
When the cell suffers low *ATP* concentration (and therefore high *ADP* concentration), *ADP* activates *Pfk* and glycolytic throughput increases. On the other hand, a cell flush with *ATP* slows glycolytic throughput as *Pfk* becomes allosterically inhibited by *ATP*. Upstream feedback activation and downstream inhibition on enzyme *Hk* works similar as the feedback on enzyme *Pfk* and it has been assumed that the feedback of *ATP* and *ADP* molecules affect the search time of the reaction.

Figure 7.5 shows the DEM representation with inhibiting feedback ν_1 from *ATP* molecules and activating feedback ν_2 from *ATP* molecules. The discrete event model can be found in Appendix E.4.

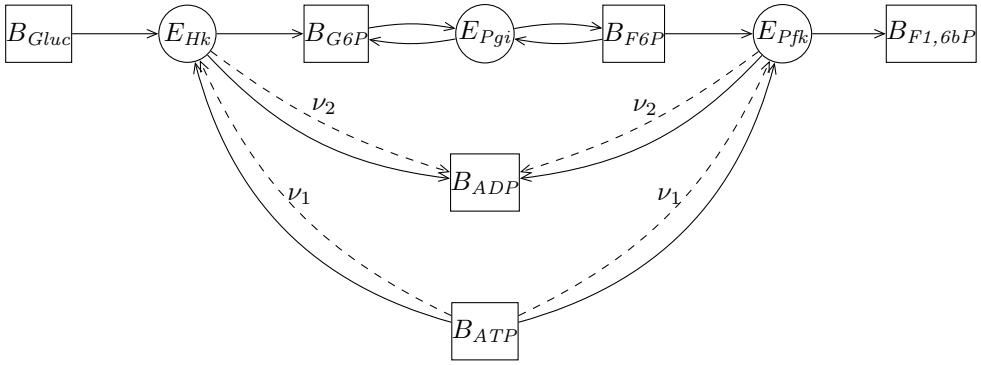

Figure 7.5: DEM representation of the EMP-pathway with feedback.

It is widely believed that when molecules bind at the regulatory site of the *Hk* or PFK enzyme their structure changes in such a way that the reconfiguration site is closed (no new substrate molecules, i.e. search process fails) or that the force of attraction towards substrate molecules increases (i.e. search process faster). This concludes that feedback molecules affect the search process. Feedback search time is calculated similar to the substrate search time. Binding time of the molecule at the regulatory site is assumed 1 second, inhibiting feedbacks pauses the search process until the inhibiting molecule unbinds and activating feedback accelerates the search process with a factor of 50. A comparison between simulations of the transient model with and without feedback is presented in Figure 7.6. Results without feedback are shown with dotted lines and initial

concentrations $C_{Gluc}(0) = 1.0$mMol and $C_{ATP}(0) = 2.0$mMol. The inhibiting effect of ATP molecules is very large at the beginning and equal to the activating feedback effect at 5 minutes. After this period activating feedback effect is dominant. This results in a slow start of the reactions but due to the activating feedback dominance after 5 minutes, the system with feedback depletes faster than the system without feedback. This also shows that the influence of feedback on this system is significant.

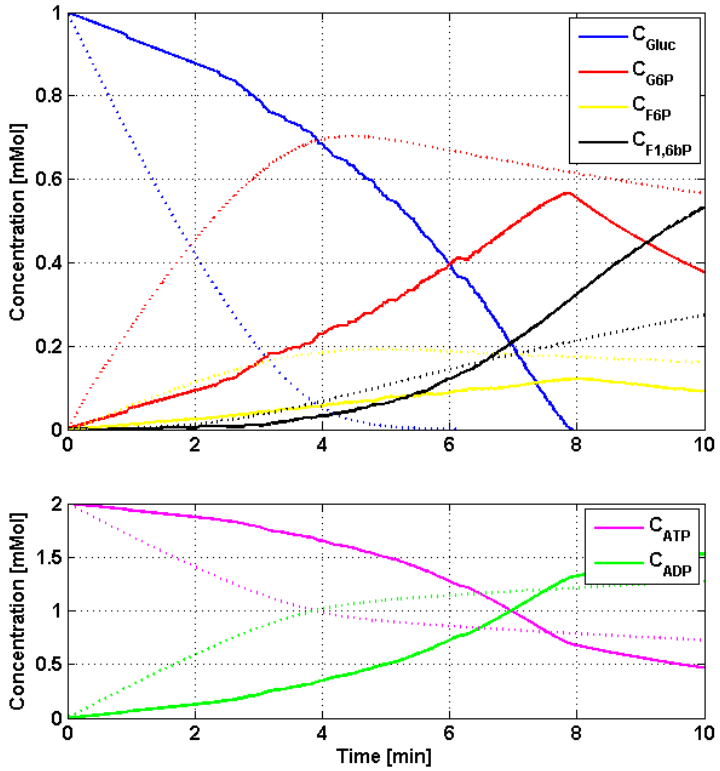

Figure 7.6: DEM results with (solid line) and without (dotted line) feedback.

7.3 Stochastic behavior

The effect of stochastic behavior of the search and reconfiguration process has been presented in this section. The search process is considered exponentially distributed and the reconfiguration process has a Γ distribution.
Simulation results of the stochastic versus the deterministic transient system are presented in Figure 7.7. The deterministic results are presented by the red dotted lines and 10 sample paths of the stochastic system are presented by the blue lines. In this case no reversible reaction or feedback is considered. The stochastic results do not differ much from the deterministic results. Increasing the coefficient of variation of the reconfiguration processes results in a larger difference between the results of the stochastic and deterministic system, see Figures 7.8 and 7.9 for a CV of respectively 9 and 30. Extremely

long reconfiguration times of enzyme *Hk* cause the wide plateaus in the last figure, like the results in [7]. A coefficient of variation of 9 or higher is not a realistic estimate for biological processes.

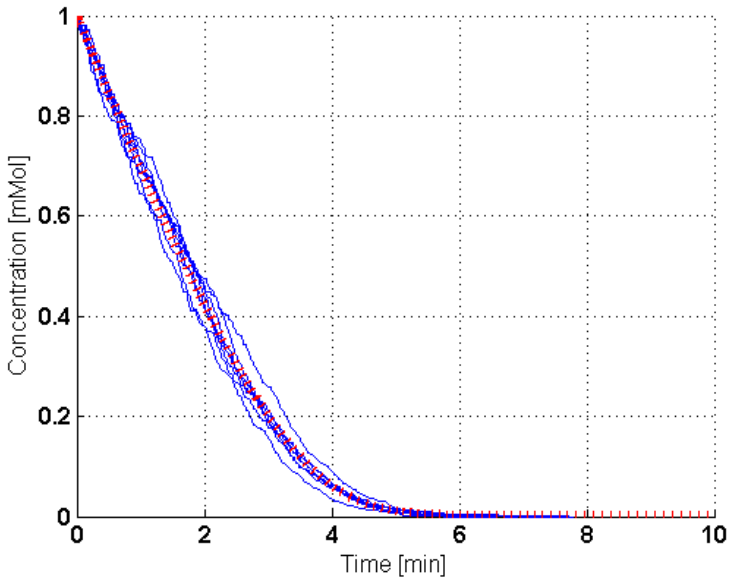

Figure 7.7: Glucose reconfiguration in transient stochastic setting with CV = 3.

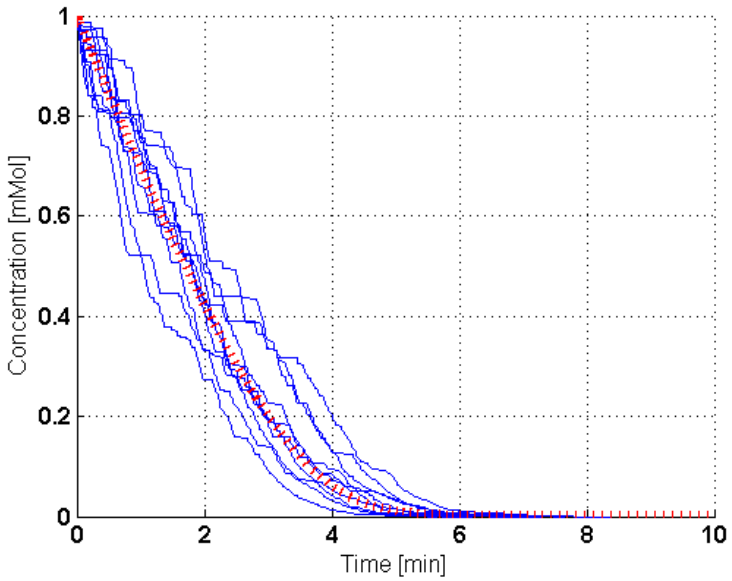

Figure 7.8: Glucose reconfiguration in transient stochastic setting with CV = 9.

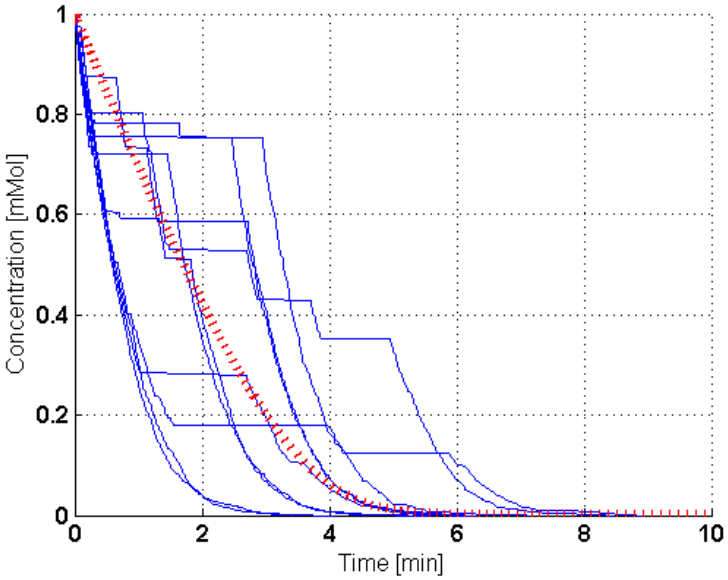

Figure 7.9: Glucose reconfiguration in transient stochastic setting with CV = 30.

Feedback of molecules also occurs in a non-deterministic way. Search time of feedback molecule and enzyme is considered exponentially distributed, similar to the search time of substrate and enzyme. When a feedback molecule binds to the regulatory site of the enzyme it unbinds after a certain period. This binding period is also considered exponentially distributed.

7.4 From transient to steady-state

The discrete event model from Figure 7.5 is a transient model. This model stops working when *ATP* has been depleted or all *Gluc* molecules are reconfigured. In the system under consideration *ATP* is consumed by enzymes *Hk* and *Pfk* while producing *ADP*, see Figure 7.1. Without *ATP* enzymes *Hk* and *Pfk* cannot reconfigure *Gluc* and *F6P* molecules.
For analyzing a steady-state simulation the DEM must be extended with an influx of *Gluc* molecules, the *ATP* has to be replenished and an outflux of *F1,6bP* molecules has to be implemented. The in- and outflux are modeled by respectively a simple generator G and an exit process X. *ATP* molecules have to be regenerated to prevent depletion. This happens outside the EMP-pathway and therefore a conversion process C has been introduced. This process represents conversion from *ADP* to *ATP*. The steady-state DEM is presented in Figure 7.10. Without *ATP* or a larger *Gluc* influx than the enzyme capacity the system becomes *unstable*, i.e. substrate molecules are piling up. When the system is *stable*, all *Gluc* molecules can be converted into *F1,6bP* molecules.

Generator G generates *Gluc* molecules with inter-arrival time t_a, conversion process C converts *ATP* to *ADP* with conversion time t_c iff there are *ADP* molecules and exit process X collects *F1,6bP* molecules with inter-exit time t_e iff there are *F1,6bP* molecules. The

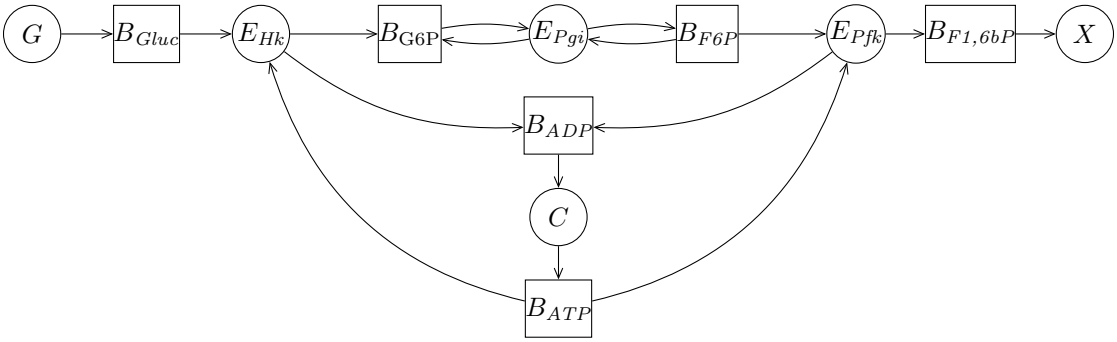

Figure 7.10: DEM representation of the steady-state EMP-pathway.

corresponding arrival speed v_{arr}, conversion speed v_{conv} and exit speed v_{exit} are:

$$v_{\mathrm{arr}} = \frac{1}{t_{\mathrm{a}}}, \tag{7.4a}$$

$$v_{\mathrm{conv}} = \frac{1}{t_{\mathrm{c}}} \qquad if\, C_{ADP} > 0, \tag{7.4b}$$

$$v_{\mathrm{exit}} = \frac{1}{t_{\mathrm{e}}} \qquad if\, C_{F1,6bP} > 0. \tag{7.4c}$$

The set of ODEs for the steady-state system are derived from (7.5) and (7.4):

$$\frac{dGluc}{dt} = -v_{Hk} + v_{\mathrm{arr}}, \tag{7.5a}$$

$$\frac{dG6P}{dt} = v_{Hk} - v_{Pgi}, \tag{7.5b}$$

$$\frac{dF6P}{dt} = v_{Pgi} - v_{Pfk}, \tag{7.5c}$$

$$\frac{dF1,6bP}{dt} = v_{Pfk} - v_{\mathrm{exit}}, \tag{7.5d}$$

$$\frac{dATP}{dt} = -v_{Hk} - v_{Pfk} + v_{\mathrm{conv}}, \tag{7.5e}$$

$$\frac{dADP}{dt} = v_{Hk} + v_{Pfk} - v_{\mathrm{conv}}. \tag{7.5f}$$

Results of both ODE and DEM system with a conversion rate of $\mu_{\mathrm{conv}} = 0.05$ mMol/minute is presented in Figure 7.11. In this simulation there was no feedback, no reversibility and no arrival or exit of molecules. It can be seen that the results are similar and *ADP* is converted into *ATP*. Figure 7.12 presents the difference between a system with *ADP* conversion and a system without *ADP* conversion. In the system without conversion not all *Gluc* and *F6P* molecules are reconfigured due to a lack of *ATP* and therefore the system stops. Figure 7.13 shows an unstable system (left figure) with $\lambda_{\mathrm{arr}} = 0.1$ mMol/minute, $\mu_{\mathrm{conv}} = 0.05$ mMol/minute and $\lambda_{\mathrm{exit}} = 0.1$ mMol/minute. This system is unstable because the arrival rate of *Gluc* molecules is higher than the conversion rate. For a stable deterministic system the arrival rate can not exceed half of the conversion rate since reconfiguration of a *Gluc* molecule to a *F1,6bP* molecule requires two *ATP* molecules. The right figure of Figure 7.12 presents a stable system with $\lambda_{\mathrm{arr}} = 0.05$ mMol/minute, $\mu_{\mathrm{conv}} = 0.1$ mMol/minute and $\lambda_{\mathrm{exit}} = 0.05$ mMol/minute. First there are start-up effects and after the start-up the concentrations do not change.

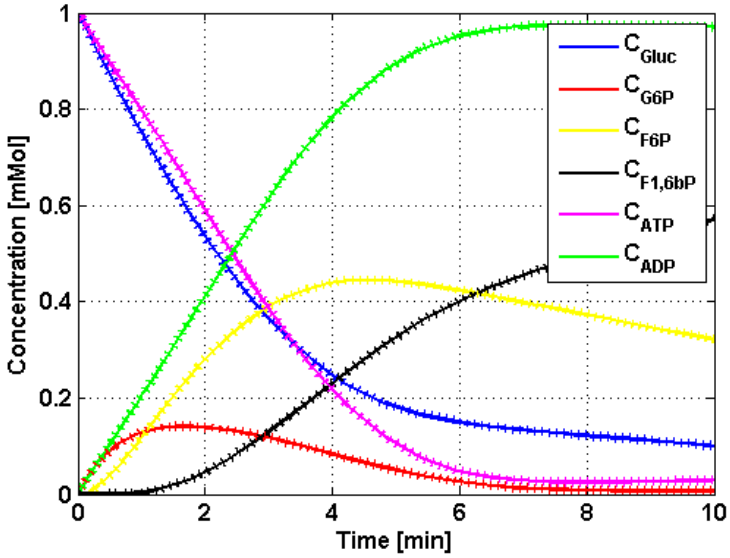

Figure 7.11: ODE (dotted line) and DEM (line) results with ADP conversion $\mu_{\text{conv}} = 0.05$ and $C_{Gluc}(0) = 1$ and $C_{ATP}(0) = 1$.

7.5 Steady-state analysis

To calculate throughput δ of the system, the number of *F1,6bP* molecules produced has been counted in a time interval. To obtain better insight, overall average throughput is scaled to mean input $\frac{1}{t_a}$ and dimensionless variable Δ in (7.6) is introduced. If $\Delta = 1$ the input is equal to the output, i.e. all incoming *Gluc* molecules will be converted into *F1,6bP* molecules. In such a system no buffers blow up, and is therefore called stable. For measurement purposes $\Delta \approx 1$ is assumed to be a stable system.

$$\Delta = \frac{\delta}{1/t_a} = \delta \cdot t_a. \tag{7.6}$$

Simulation results of the steady-state system with different inter-arrival t_a and conversion t_c times are presented in Table 7.1. The concentrations are presented in μMol and initial concentrations of ATP and ADP are both 100 μMol. Average concentrations are calculated over the time period of 1000-2000 minutes, after the start-up phase. For the stochastic results an average concentration of 20 simulations is taken. Average concentrations can also be calculated by hand, knowing both $V_{\max} = 1/t_a$ and the ATP concentration. These results are also presented in Table 7.1. It can be seen that the stochastic results do not differ much from the deterministic results. Simulation results with reversible reaction (Rev) and feedback (FB) are presented. The results of the model with reversible reaction are similar to a system without reversibility, except for the $G6P$ concentration which is much higher due to the reconfiguration of $G6P$ back into $F6P$. The feedback model with an arrival time of 0.009 and conversion time of 0.001 is unstable. Since the reconfiguration rate is very high, the ATP concentration is almost maximal and ADP concentration almost minimal. Therefore, lots of inhibiting and almost no activating feedback molecules bind with the enzyme. For an arrival time close to the minimal reconfiguration time this system cannot handle the influx and is unstable.

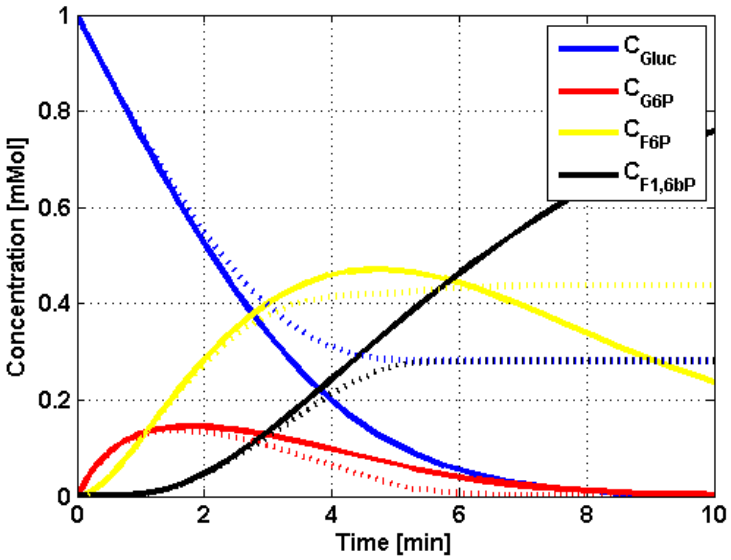

Figure 7.12: DEM results with (line) and without (dotted line) *ADP* conversion with $\mu_{\text{conv}} = 0.1$, $C_{Gluc}(0) = 1$ and $C_{ATP}(0) = 1$.

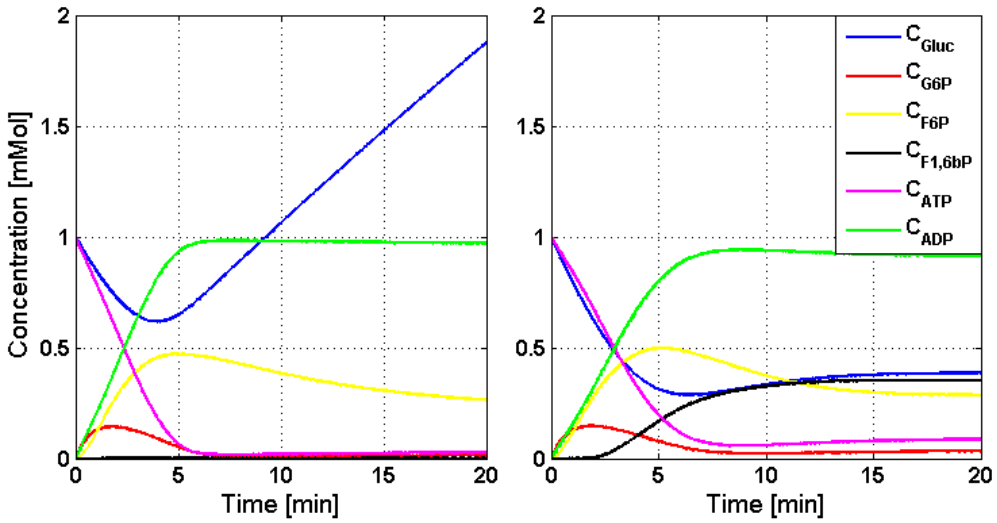

Figure 7.13: DEM results of an unstable (left) and stable (right) system with $C_{Gluc}(0) = 1$ and $C_{ATP}(0) = 1$.

Table 7.1: Comparison between simulations of a steady-state system.

t_a	t_c	*Gluc*	*G6P*	*F6P*	*ATP*	*ADP*
0.009	0.001					
Det (hand)		868.0	73.5	487.9	200.0	0.0
Det		868.0	73.0	487.1	199.1	0.0
Stoch		891.4	73.5	490.1	199.0	0.1
Difference (%)		2.7	0.7	0.6	0.1	
Det (Rev)		868.0	1760.0	487.1	199.1	0.0
Stoch (Rev)		891.2	1785.0	488.5	199.0	0.1
Det (FB)				Unstable		
0.016	0.007					
Det (hand)		117.3	40.9	292.3	200.0	0.0
Det		118.0	40.0	292.0	198.6	0.4
Stoch		124.7	40.9	293.7	193.2	5.6
Difference (%)		5.7	2.3	0.6	-2.7	1300
Det (Rev)		118.0	1100.0	292.0	198.0	0.0
Stoch (Rev)		124.5	1037.8	293.6	193.3	5.4
Det (FB)		218.7	19.3	190.9	66.6	132.0
0.016	0.008					
Det (hand)		117.3	40.9	292.3	200.0	0.0
Det (hand)		665.3	40.9	329.4	100.0	100.0
Det		118.0	40.0	292.0	198.4	0.2
Stoch		360.0	40.9	315.4	128.9	69.7
Difference (%)		205.1	2.3	8.0	-35.0	34750
Det (Rev)		118.0	1100.0	292.0	198.8	0.0
Stoch (Rev)		332.2	1108.5	314.0	198.2	0.6
Det (FB)		361.2	18.9	142.8	5.5	193.1

7.6 Stability boundary

To analyze the stability boundary, simulations with different arrival and conversion rates are performed. The model used is the model of the steady-state EMP-pathway without reversibility and feedback. Results are presented as scaled outflux Δ as a function of conversion time t_c and arrival time t_a in Figure 7.14. Each point in the figure is the mean value of 5 simulations (CV < 0.004). Also a deterministic value for Δ is added, and can be calculated by (7.7). Enzyme *Hk* is the bottleneck of this system and with maximal *ATP* concentration of 200μmol maximal conversion time t_{Gluc} is 0.0079 minutes. In the figure three regimes can be distinguished:

- *Overloaded system due to slow regeneration of* ATP. For each *Gluc* molecule entering the system and exiting as $F16bP$ molecule, two *ATP* molecules are reconfigured into *ADP* molecules. Therefore, *ATP* regeneration must be at least twice as fast as the arrival rate to fulfill the need for *ATP*. For $t_c/t_a \leq 0.5$, Δ decreases below 1.
- *Stable system.* $t_c/t_a \leq 0.5$ and $t_a < 0.0085$. For these parameters there is nu pile-up in the buffers and the outflux is equal to the influx.
- *Increasing the glucose influx beyond enzyme capacity.* For $t_a < 0.0085$ the incoming *Gluc* molecules can not all be reconfigured and $\Delta < 1$, due to stochastic influence and minimal reconfiguration time t_{Gluc} of 0.0079 minutes

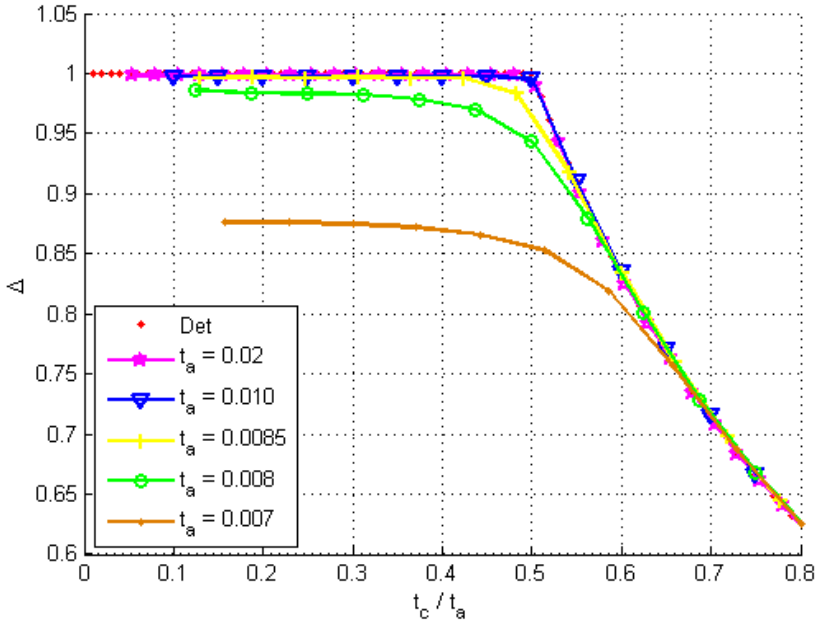

Figure 7.14: Stability boundary of the system without reversibility and feedback.

$$\Delta = \begin{cases} 1 & \text{if } t_a \geq 2t_c \text{ and } t_{Gluc} \geq t_a \\ t_a/2t_c & \text{if } t_a < 2t_c \text{ and } t_{Gluc} \geq t_a \\ t_a/(max(2t_c, t_{Gluc}) & \text{if } t_{Gluc} < t_a \end{cases} \tag{7.7}$$

The stability boundary of the system with feedback and reversible reaction is presented in Figure 7.15. It can be seen that in this figure the same regimes can be distinguished. Due to the feedback and reversibility t_{Gluc} has increased, what results in a lower arrival rate for a stable system.

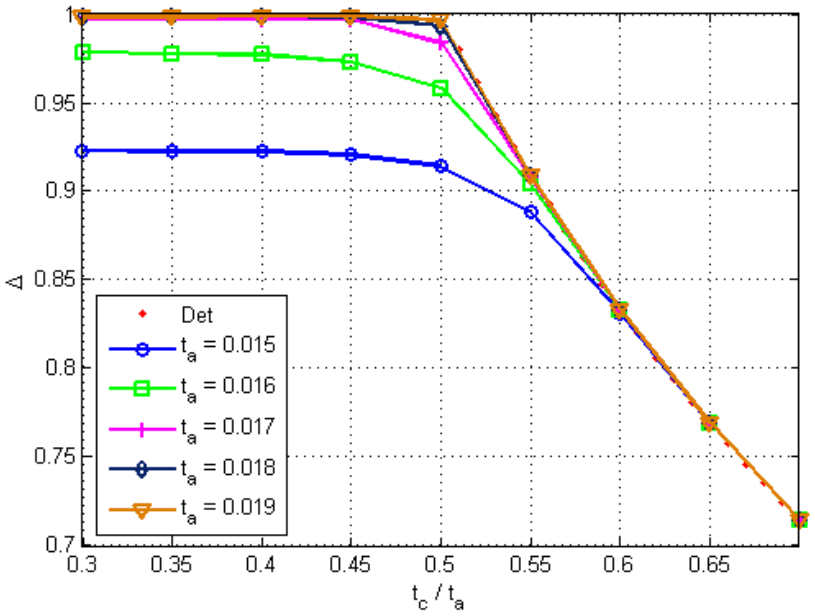

Figure 7.15: Stability boundary of the system with reversibility and feedback.

7.7 Conversion related to ADP concentration

In the steady-state model from Section 7.4 the conversion of ADP to ATP molecules was assumed to be at a fixed rate. A more realistic approach for this conversion is to let the conversion rate depend on the concentration of ADP molecules. In other words, if the ADP concentration is very low conversion to ATP will be slow and if the ADP concentration is large conversion to ADP will be faster. We assumed a linear relation between ADP concentration and conversion rate:

$$\mu_{\text{conv}} = \mu_{\text{c,max}} \cdot \frac{[ADP]}{[ADP]_0 + [ATP]_0}, \tag{7.8}$$

The ADP concentration is divided by the maximal ADP concentration $[ADP]_0 + [ATP]_0$ to achieve that if the concentration is maximal, the conversion rate will also be maximal, see Figure 7.16. In the set of ODEs (7.4b) must be changed into (7.9).

$$v_{\text{conv}} = \frac{1}{t_{\text{c,max}}} \cdot \frac{[ADP]}{[ADP]_0 + [ATP]_0}, \tag{7.9}$$

Simulation results of the system with conversion depending on the ADP concentration are presented in Figure 7.17.

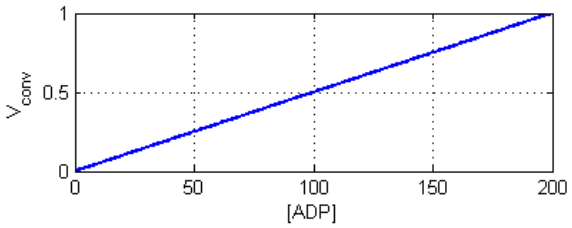

Figure 7.16: Conversion speed related to the ADP concentration.

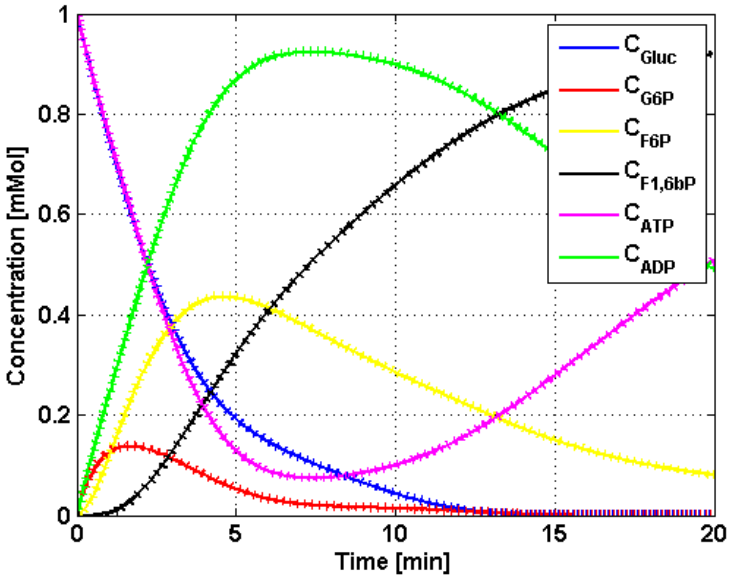

Figure 7.17: ODE and DEM results of the system with conversion depending on the ADP concentration, $t_{c,\max} = 0.01$, $C_{Gluc}(0) = 1$ and $C_{ATP}(0) = 1$.

7.8 Conclusion

The EMP-pathway has been modeled as a discrete event model, extended with reversibility and feedback of ATP and ADP molecules. These transient models are simulated, verified and finally expanded to steady-state models. The results of these models are analyzed and the everything seems to work according to the desired behavior. Finally, we assumed that the conversion of ADP molecules did not occur at a fixed rate, but is linear dependent on the ADP concentration. As the ADP concentration grows larger, conversion rate increases and vice versa.

Chapter 8

Conclusions and recommendations

Modeling substrate-enzyme reactions with discrete event models has been described in this report. First the substrate-enzyme reaction has been simplified to one reaction; substrate S binds with enzyme E, the enzyme reconfigures the substrate into product P and unbinds. This reaction has been modeled with a set of ODEs. For modeling the reaction as a discrete event model, using the manufacturing approach, it has been divided into two distinct processes. First, the *search* process describes the time it takes between a substrate molecule and enzyme molecule to bind, depending on the number of molecules and corresponds to the setup time of a machine. Second, the *reconfiguration* process describes the reconfiguration of the substrate into a product molecule and unbinding from the enzyme and corresponds to the processing time of a machine.
Results of simulations with the deterministic DEM of the substrate-enzyme reaction have been verified with the ODEs. An important difference between the commonly used ODEs and the DEM used in manufacturing lines is that the reaction has been divided over two distinct processes. This division is important when using stochastic distributions since it is not clear how the reconfiguration process is distributed. Since an arbitrary distribution can be chosen for the search or reconfiguration process in the DEM, essentially any biophysical hypothesis providing the details of the process can be implemented in the system. Biologists use exponential distributions for the total reaction but experiments show that this is not (always) the case.

This basic model has been extended with inhibiting and/or activating feedback. The enzyme has, along with the catalytic site where substrates are reconfigured, a regulatory site where upstream substrate and/or downstream product molecules can bind. These molecules act as feedback molecules when bound at the regulatory site, affecting the reaction speed. Feedback on the enzyme can influence both search or reconfiguration process. If the search process is influenced, no substrate molecules can bind at the enzyme while an inhibiting feedback molecule is bound at the regulatory site and while an activating feedback molecule is bound the attraction between a substrate molecule and the reconfiguration site increases. The reconfiguration process is influenced by increasing or decreasing the reconfiguration rate while respectively an activating or inhibiting feedback molecule is bound to the regulatory site of the enzyme.

A combination of an enzyme with feedback and a basic enzyme both competing for

the same substrate molecules resulted in a activity function similar to a decay rate that is experimentally found in literature, but hasn't been well defined.

The basic substrate-enzyme model has also been extended into a reversible reaction, i.e. product molecules can also be reconfigured back into substrate molecules. The deterministic reversible DEM uses the overall reaction rate, while the stochastic model uses both forward and backward reaction rates. This is to make sure stochastic fluctuations occur, which would not be the case if the reaction is in equilibrium using the overall reaction rate. The deterministic DEM has been verified with a model of ODEs.

With these enzymes the first steps of glycolysis, the archetype of a universal metabolic pathway, have been modeled. This is model consisting of three reactions. The DEM model has been verified with an ODE model. Steady-state simulations have been conducted to compare the deterministic and stochastic models. With these settings there has been no significant difference. Analysis of the stability boundary showed three distinct regimes depending on the arrival rate of new substrates and the conversion between energy molecules.

The χ models used for the DEM simulations have been constructed from small processes. With correct linking between these processes other pathways and reactions can be modeled and simulated.

Extension of the current steps of the glycolysis pathway with the current processes is a topic for further research. The models representing substrate-enzyme reactions can also be used as building blocks for other pathways.

In order to check if the DEM represents real-life behavior, comparison with experimental results is necessary. Also simplifications and assumptions in the model, like using Michaelis-Menten equations can be a topic for further research.

Bibliography

[1] FI Ataullakhanov and VM Vitvitsky . What determines the intracellular atp concentration. *Biosci Rep.*, 22(5):501–511, December 2002.

[2] G.E. Briggs and J.B.S. Haldane. A Note on the Kinetics of Enzyme Action. *Biochem. Journal*, 19:338–339, 1925.

[3] Daniel T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22:403–434, December 1976.

[4] Nobu Ishiia and Yoshi Sugaa. Dynamic simulation of an in vitro multi-enzyme system. 2007.

[5] S. C. Kou and B.J. Cherayil. Single-Molecule Michaelis-Menten Equations. 2005.

[6] L. Michaelis and M. Menten. Die Kinetik der Invertinwirkung. *Biochemische Zeitschrift*, 49:333–369, 1913.

[7] E. A. F. van de Rijt. Discrete event simulations for biological systems. Master's thesis, Eindhoven University of Technology.

[8] J. Vervoort and J.E. Rooda. Learning timed χ 1.0. 2007.

[9] D. J. Wilkinson. *Stochastic modelling for systems biology.* Chapman and Hall/CRC Press, Boca Raton, Florida, 2006.

Appendix A

Parameters

Table A.1: Parameters and initial concentrations from [9].

Parameter	Value
$[S]_0$	$5.0{\cdot}10^{-7}$
$[E]_0$	$2.0{\cdot}10^{-7}$
$[P]_0$	0
k_1	$1.0 \cdot 10^{6}$
k_2	$1.0 \cdot 10^{-4}$
k_3	0.1
K_m	$1.001 \cdot 10^{-7}$

Table A.2: EMP parameters based on [4].

Enzyme	Parameter	Value	
Hk	C_{Hk}	0.05	μMol
	$V_{\max}$	225	μMol·min^{-1}·mg^{-1}
	$K_{\mathrm{m,Gluc}}$	0.12	mMol
	$K_{\mathrm{m,ATP_1}}$	0.50	mMol
	$k_{\mathrm{Hk,inactivation}}$	0.29	min^{-1}
	$K_{\mathrm{eq,Hk,inactivation}}$	2.72	
DEM	$V_{\max}$	442.63982475	$\tau_r = \frac{1}{V_{\max}}$
Pgi	C_{Pgi}	0.05	μMol
	$V_{\max}$	1511	μMol·min^{-1}·mg^{-1}
	$K_{\mathrm{m,G6P}}$	3.0	mMol
	$K_{\mathrm{m,F6P}}$	0.16	mMol
	$K_{\mathrm{eq,Pgi}}$	0.30	
DEM	$V_{\max}$	4647.60935	$\tau_r = \frac{1}{V_{\max}}$
Pfk	C_{Pfk}	0.05	μMol
	$V_{\max}$	145	μMol·min^{-1}·mg^{-1}
	$K_{\mathrm{m,F6P}}$	0.46	mMol
	$K_{\mathrm{m,ATP_2}}$	0.04	mMol
	n	1.9	
DEM	$V_{\max}$	252.5465	$\tau_r = \frac{1}{V_{\max}}$

Table A.3: Specific weight of enzymes.

Enzyme	gr/Mol
Hk	34717
Pgi	61517
Pfk	34834

Appendix B

Types

The only type used in the Chi models is the molecule, it consists of the id number and the time it entered the system:

```
type mol = ( nat, real ) // id, timein
```

Appendix C

Functions

All functions used for the DEM in this report are presented in this appendix with a small description.

C.1 *injBuff*

Function `injBuff` injects new molecule `x` in list `s` according to a value from uniform distribution `p`, and returns the new list:

```
func injBuff( val xs: [mol], x: mol, p: real ) -> [mol] =
|[ var s: nat
:: s:= i2n ( floor (( len(xs) + 1) * p ))
 ; xs:= take (xs, s) ++ [x] ++ drop (xs, s)
 ; ret xs
]|
```

C.2 *meanST*

Function `meanST` calculates the mean search time with values $K_{\mathrm{m,S}}$, $K_{\mathrm{m,ADP}}$ and τ_r of the enzyme and the buffer content `blevel`:

```
func meanST( val km, tr: real, blevel : nat ) -> real =
|[ ret (( km + blevel ) / blevel - 1 ) * tr ]|
```

C.3 *meanST1*

Function `meanST` calculates the mean search time with values K_m and τ_r of the enzyme and the buffer contents of the substrate and ATP molecules `conc1` and `conc2`:

```
func meanST1( val km1, km2, tr, n: real , conc1,conc2 : nat ) -> real =
|[ ret ((( km1^n + conc1^n )*( km2 + conc2 )) / ( conc1^n * conc2 ) - 1 ) * tr ]|
```

C.4 *calcVpgi*

Function `calcVpgi` calculates specific activity without reversibility:

```
func calcVpgi ( val kms,kmp,keq,Vmax: real , concS,concP: nat ) -> real =
|[ ret ( Vmax * concS ) / ( kms + concS ) ]|
```

C.5 *calcVpgiRev*

Function `calcVpgi` calculates specific activity with reversibility:

```
func calcVpgi ( val kms,kmp,keq,Vmax: real , concS,concP: nat ) -> real =
|[ ret ( Vmax * (concS - concP/keq)) / (kms * (1 + concP/kmp) + concS) ]|
```

C.6 *cond*

Function `cond` calculates if the upstream or downstream reconfiguration rate is dominant:

```
func cond ( val Vpgi: real, concS,concP: nat ) -> int =
|[ ( Vpgi < 0.0 and concP > 0 -> ret -1
   | Vpgi = 0.0 or Vpgi < 0.0 and concP = 0 -> ret +0
   | Vpgi > 0.0 and concS = 0 -> ret +0
   | Vpgi > 0.0 and concS > 0 -> ret +1
   )
]|
```

Appendix D

Processes

The Chi models used in this report consist of several modules. These modules, buffers enzymes and functions are presented in this appendix. Appendix E presents the complete chi models.

D.1 Buffers

Basic buffer B

This is a basic buffer, it can receive molecules at all times via channel `a` and if the buffer is non-empty it can send molecules via channel `b`. Chi file:

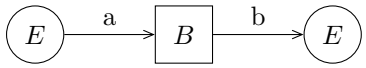

Figure D.1: DEM representation of buffer B.

```
proc B( chan a?,b!: mol, dt!: (string,nat), val S0: nat, id: string ) =
|[ var xs: [mol] = []
     , x: mol
     , uni: -> real = uniform ( 0.0 , 1.0 )
     , i: nat = 1
:: i <= S0
   *> ( xs:= injBuff ( xs, (i,time), sample uni ); i:= i + 1 )
 ; *( dt !!( id , len (xs))
```

```
      ; ( a?x; xs:= injBuff ( xs, x, sample uni)
        | len (xs) > 0 -> b!hd(xs); xs := tl(xs)
        )
    )
]|
```

Model input:

```
|| B(a,b,dt,S0,id)
```

D.1.1 Communication with one enzyme *B1*

This buffer sends its bufferlength when changed to upstream or downstream enzymes over channel `r` channel.

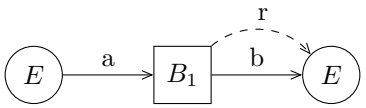

Figure D.2: DEM representation of buffer *B1*.

Chi file:

```
proc B1( chan a?,b!: mol, r!: nat, q!:void, p?: bool
       , dt!: (string,nat), val S0: nat, id: string
       ) =
|[ var xs: [mol] = []
     , x: mol
     , uni: -> real = uniform ( 0.0 , 1.0 )
     , i: nat = 1
     , search: bool = false
:: i <= S0
   *> ( xs:= injBuff ( xs, (i,time), sample uni ); i:= i + 1 )
 ; r!len(xs)
 ; *( dt !!( id , len (xs))
      ; ( a?x; xs:= injBuff ( xs, x, sample uni)
        | len (xs) > 0 -> b!hd(xs); xs := tl(xs)
        )
      ; r!len(xs)
      ; ( search     -> q!!
        | not search -> skip
        )
    )
|| *p?search
]|
```

Model input:

```
|| B1(a,b,r,dt,S0,id)
```

D.1.2 Communication with two enzymes *B2*

This buffer is almost similar to Figure D.2 but instead of communicating with one enzyme, it can communicate with both the upstream and downstream enzyme, see Figure D.3.

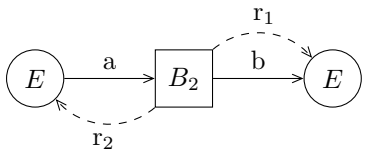

Figure D.3: DEM representation of buffer *B2*.

Chi file:

```
proc B2( chan a?,b!: mol, r1!,r2!: nat, q1!,q2!:void, p1?,p2?: bool
       , dt!: (string,nat), val S0: nat, id: string
       ) =
|[ var xs: [mol] = []
     , x: mol
     , uni: -> real = uniform ( 0.0 , 1.0 )
     , i: nat = 1
     , search1,search2: bool =  ( false, false )
:: i <= S0
   *> ( xs:= injBuff ( xs, (i,time), sample uni ); i:= i + 1 )
 ; r1!len(xs); r2!len(xs)
 ; *( dt !!( id , len (xs))
      ; ( a?x; xs:= injBuff ( xs, x, sample uni)
        | len (xs) > 0 -> b!hd(xs); xs := tl(xs)
        )
      ; r1!len(xs); r2!len(xs)
      ; ( search1     -> q1!!
        | not search1 -> skip
        )
      ; ( search2     -> q2!!
        | not search2 -> skip
        )
    )
|| *( p1?search1 | p2?search2 )
]|
```

Model input:

```
|| B2(a,b,r1,r2,dt,S0,id)
```

D.1.3 Exit *BX*

In Figure D.4 a buffer is presented which has incoming channel `a`.

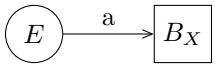

Figure D.4: DEM representation of buffer *BX*.

Chi file:

```
proc BX( chan a?: mol, dt!: (string,nat), val id: string ) =
|[ var xs: [mol] = [], x: mol, i: nat = 0
:: *( dt!!( id , i ); a?x; i:= i + 1 )
]|
```

Model input:

```
|| BX(a,dt,id)
```

D.2 Enzymes

D.2.1 Basic, one molecule *E1*

This process represents an enzyme that receives a substrate molecule via channel `a` after a search time calculated with function `meanST`. The enzyme reconfigures the substrate and sends the product molecule via channel `b`. Communication about bufferlength with the upstream buffer is via channel `r`. The DEM representation of this process is presented in Figure D.5.

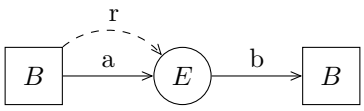

Figure D.5: DEM representation of process *Enz1*.

Chi file:

```
proc E( chan a?,b!: mol, r?: nat, q?: void, p!:bool
           , val Vmax, km, tr: real
      ) =
|[ var x: mol
     , searchStart , delayTimeS, delayTimeP: real
     , concS: nat
```

```
     , search: bool = false
:: *( concS > 0 -> searchStart := time
      ; delayTimeS := ( searchStart + meanST( km, tr, concS )
                      - time ) max 0.0
      ; search:= true; p!search
      ; search
        *> ( delay delayTimeS; search:= false
           | q?
             ; ( concS > 0 -> delayTimeS := ( searchStart +
                   meanST( km, tr, concS ) - time ) max 0.0
               | concS = 0 -> search:= false
               )
           )
      ; p!search
      ; ( concS > 0 -> a?x
          ; delayTimeP:= 1 / Vmax
          ; delay delayTimeP
          ; b!x
        | concS = 0  -> skip
        )
    )
|| * r?concS
]|
```

Model input:

```
|| E1(a, b, r, Vmax, Km, tr)
```

D.2.2 Basic, two molecules *E2*

This process is almost similar to the basic enzyme with one molecule, but it requires two different substrate molecules befor it can start the reconfiguration process, the model is presented in Figure D.6.

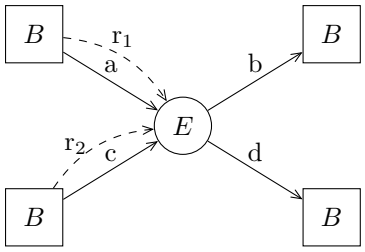

Figure D.6: DEM representation of process *E2*.

Chi file:

Model input:

```
|| E2(a,b,c,d,r1,r2,Vmax,km1,km2,tr,n)
```

D.2.3 Feedback search process (inhibiting) *Efb_st_inh*

Up- or downstream inhibiting feedback affecting the search process. The DEM representation of this feedback process is presented in Figure D.7.

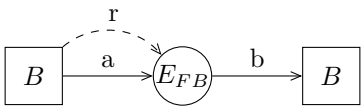

Figure D.7: DEM representation of process *Efb_st_inh and Efb_st_act.*

Chi file:

```
proc Efb ( chan a?,b!: mol, r?: nat, f?,q?: void
          , t!,p!:bool, val Vmax,km,tr: real
          ) =
|[ var x: mol
     , procTime, ts, tStart : real
     , boundTime: real = 1/60
     , concS: nat = 0
     , search, processing, fb: bool = ( false, false, false )
:: *( concS > 0 and fb= false -> tStart:=time
      ; search:= true
      ; p!search
      ; search
        *> ( ts := ( tStart + meanST( km, tr, concS )
                            - time ) max 0.0
             ;( delay ts; search:= false
              | q?
                ; ( concS > 0 -> skip
                  | concS = 0 -> search:= false
                  )
              )
        )
      ; p!search
      ; ( concS > 0  -> a?x
          ; procTime:= 1/Vmax
          ; delay procTime
          ; b!x
        | concS = 0 -> skip
        )
    )
|| * r?concS
|| *( f?
      ; fb:= true
      ; t!fb
      ; delay boundTime
      ; fb:= false
      ; t!fb
    )
]|
```

Model input:

```
|| Efb(a, b, r, Vmax, Km, tr)
```

D.2.4 Feedback search process (activating) *Efb_st_act*

Up- or downstream activating feedback affecting the search process. The DEM representation of this feedback process is presented in Figure D.7.

Chi file:

```
proc Efb ( chan a?,b!: mol, r?: nat, f?,q?: void
         , t!,p!:bool, val Vmax,km,tr: real
         ) =
|[ chan u: void
   ,var x: mol
      , procTime, ts, tStart : real
      , boundTime: real = 1/60
      , concS: nat = 0
      , search, processing, fb: bool = ( false, false, false )
:: *( concS > 0 and fb= false -> tStart:=time
      ; search:= true
      ; p!search
      ; search
        *> ( ts := ( tStart + meanST( km, tr, concS )
                            - time ) max 0.0
             ;( delay ts; search:= false
              | (q?|u?)
                ; ( concS > 0 -> skip
                  | concS = 0 -> search:= false
                  )
              )
        )
      ; p!search
      ; ( concS > 0  -> a?x
          ; procTime:= 1/Vmax
          ; delay procTime
          ; b!x
        | concS = 0 -> skip
        )
    )
|| * r?concS
|| *( f?
      ; tr:= tr / 50
      ; t!true
      ; ( search     -> u!
        | not search -> skip
        )
      ; delay boundTime
      ; tr:= tr * 50
      ; ( search     -> u!
        | not search -> skip
        )
```

```
      ; t!false
    )
]|
```

Model input:

```
|| Efb(a, b, r, Vmax, Km, tr)
```

D.2.5 Feedback search process *Efb_st*

This enzyme processes two molecules, with an upstream activating feedback and downstream inhibiting effect. The feedback influences the search process. The DEM representation of this feedback process is presented in Figure D.8.

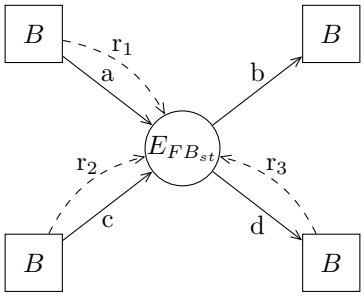

Figure D.8: DEM representation of process *Efb_st*.

Chi file:

```
proc Efb2( chan a?,b!,c?,d!: mol, r1?,r2?: nat, f1?,f2?,q1?,q2?: void
          , t1!,t2!,p1!,p2!:bool, val Vmax,km1,km2,tr,n: real
         ) =
|[ chan u: void
  , var x1,x2: mol
      , tStart, ts, tp, mst, un: real
      , boundTime: real = 1/60
      , concS1,concS2: nat = ( 0, 0 )
      , search, Inh, Act: bool = ( false, false, false )
:: *( concS1 > 0 and concS2 > 0 and Inh = false -> tStart:= time
      ; search:= true
      ; p1!search; p2!search
      ; search
        *> ( ts := ( tStart + meanST1(km1,km2,tr,n,concS1,concS2)
                            - time ) max 0.0
             ;( delay ts; search:= false
              | ( q1? | q2? | u? )
                ; ( concS1 > 0 and concS2 > 0 -> skip
                  | concS1 = 0 or concS2 = 0 -> search:= false
                  )
              )
           )
```

```
      ; p1!search; p2!search
      ;( concS1 > 0 and concS2 > 0  -> a?x1 ; c?x2
         ; tp:= 1 / Vmax
         ; delay tp
         ; b!x1; d!x2
       | concS1 = 0 or concS2 = 0  -> skip
       )
    )
|| *( r1?concS1 | r2?concS2 )
|| *( ( f1?; Inh:= true
      | f2?; Act:= true; tr := tr / 50
        ;( search     -> u!!
         | not search -> skip
         )
      )
      ; t1!true; t2!true
      ; delay boundTime
      ; ( Inh -> Inh := false
        | Act -> Act := false;  tr := tr * 50
          ;( search     -> u!
           | not search -> skip
           )
        )
      ; t1!false; t2!false
    )
]|
```

Model input:

```
|| Efb2(a,b,c,d,r1,r2,r3,Vmax,km1,km2,tr,n)
```

D.2.6 Feedback reconfiguration process *Efb_r*

The process of Section D.2.1 is extended with activating feedback from an upstream buffer and with inhibiting feedback from a downstream buffer. The feedback molecules affect the reconfiguration process. The DEM representation of this feedback process is presented in Figure D.9.

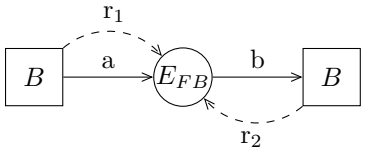

Figure D.9: DEM representation of process *Enz_r*.

Chi file:

```
proc Efb( chan a?,b!: mol, r?: nat, nuS?,nuP?,q?: void
            , tS!,tP!,p!:bool, val Vmax, km, tr: real
```

```
        ) =
|[ var x: mol
     , searchStart , delayTimeS, readyTimeP : real
     , procStart , delayTimeP, boundStart : real
     , boundTime: real = 0.1
     , concS: nat = 0
     , processing, search: bool = ( false, false )
     , bindS, bindP, newS, newP: bool = ( false, false, false, false )
:: *( concS > 0 -> searchStart := time
      ; search:= true; p!search
      ; search
        *> ( delayTimeS := ( searchStart + meanST( km, tr, concS )
                             - time ) max 0.0
             ; ( delay delayTimeS; search:= false
               | q?
                 ; ( concS > 0 -> skip
                   | concS = 0 -> search:= false
                   )
               )
           )
      ; p!search
      ; ( concS > 0 -> a?x
          ; procStart:= time
          ; readyTimeP:= procStart + 1 / Vmax
          ; newS:= true; newP:= true
          ; processing:= true
          ; processing
            *> ( delayTimeP:= ( readyTimeP - time ) max 0.0
                 ;( delay delayTimeP; processing:= false
                  | bindS and newS ->
                      readyTimeP:= time + ( readyTimeP - time ) / 2
                    ; newS:= false
                  | not bindS and not newS ->
                      readyTimeP:= time + ( readyTimeP - time ) * 2
                    ; newS:= true
                  | bindP and newP ->
                      readyTimeP:= time + ( readyTimeP - time ) * 2
                    ; newP:= false
                  | not bindP and not newP ->
                      readyTimeP:= time + ( readyTimeP - time ) / 2
                    ; newP:= true
                  )
               )
          ; b!x
        | concS = 0  -> skip
        )
   )
|| * r?concS
|| *( ( nuS?; boundStart:= time; bindS:= true
      | nuP?; boundStart:= time; bindP:= true
      )
      ; tS!true; tP!true
      ; delay boundTime
      ; bindS:= false; bindP:= false
      ; tS!false; tP!false
```

```
    )
]|
```

Model input:

```
|| Efb(a, b, r1, r2, Vmax, Km, tr)
```

D.2.7 Feedback reconfiguration process, 2 substrates *Efb2*

The process of Section D.2.6 is extended with an extra incoming and outgoing channel. The DEM representation of this feedback process is presented in similar to the representation in Figure D.8.

Chi file:

```
proc Efb2( chan a?,b!,c?,d!: mol, r1?,r2?: nat, f1?,f2?,q1?,q2?: void
          , t1!,t2!,p1!,p2!:bool, val S1,S2: nat, Vmax,km1,km2,tr,n: real
          ) =
|[ var subs,atp: mol
     , searchStart , delayTimeS : real
     , procStart , delayTimeP: real
     , boundTime: real = 0.1
     , concS,concATP: nat = (S1,S2)
     , search, processing: bool = ( false, false )
     , bindS, bindP, newS, newP: bool = ( false, false, false, false )
:: *( concS > 0 and concATP > 0 -> searchStart := time
      ; search:= true; p1!search; p2!search
      ; search
        *> ( delayTimeS := ( searchStart + meanST1(km1,km2,tr,n
                           ,concS,concATP) - time ) max 0.0
             ;( delay delayTimeS; search:= false
              | ( q1? | q2? )
                ; ( concS > 0 and concATP > 0 -> skip
                  | concS = 0 or  concATP = 0 -> search:= false
                  )
              )
           )
       ; p1!search; p2!search
       ; ( concS > 0 and concATP > 0  -> a?subs ; c?atp
           ; procStart:= time
           ; delayTimeP:= ( procStart + 1 / Vmax   - time ) max 0.0
           ; newS:= true; newP:= true
           ; processing:= true
           ; processing
             *> ( delay delayTimeP; processing:= false
                | bindS and newS ->
                  delayTimeP:= ( (delayTimeP + procStart - time) / 2 ) max 0.0
                  ; newS:= false
                | not bindS and not newS -> delayTimeP:=
                  ( (delayTimeP + procStart - time) * 2 ) max 0.0
                  ; newS:= true
                | bindP and newP ->
```

```
                  delayTimeP:= ( (delayTimeP + procStart - time) * 2) max 0.0
                  ; newP:= false
                | not bindP and not newP ->
                  delayTimeP:= ( (delayTimeP + procStart - time) / 2 ) max 0.0
                  ; newP:= true
                )
            ; b!subs; d!atp
          | concS = 0 or concATP = 0 -> skip
          )
    )
|| *( r1?concS | r2?concATP )
|| *( ( f1?; bindS:= true
      | f2?; bindP:= true
      )
      ; t1!true; t2!true
      ; delay boundTime
      ; bindS:= false; bindP:= false
      ; t1!false; t2!false
    )
]|
```

Model input:

```
|| Efb2(a,b,c,d,r1,r2,r3,Vmax,km1,km2,tr,n)
```

D.2.8 Reversible *Erev*

The reversible reaction is presented in Figure D.10. The molecules flow in the direction calculated by function `cond` with a speed calculated by `calcVpgiRev`.

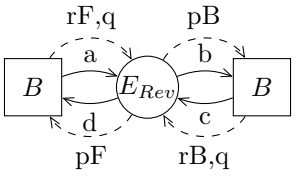

Figure D.10: DEM representation of process *Erev*.

Chi file:

```
proc Erev( chan a?,b!,c?,d!: mol, rF?,rB?: nat, q?: void, pF!,pB!:bool
             , val Vmax,kmF,kmB,keq,tr: real
         ) =
|[ var x: mol
     , procTime: real
     , Vpgi,  searchStart , delayTimeS : real
     , concS , concP: nat = ( 0, 0 )
     , search: bool = false
```

```
     , side: int = +0
:: *( concS /= concP / keq -> Vpgi:= calcVpgi( kmF,kmB,keq,Vmax,concS,concP)
      ; side:= cond( Vpgi, concS, concP )
      ; searchStart := time
      ; search:= true; pF!search; pB!search
      ; search
        *> ( delayTimeS := ( searchStart + (Vmax / ( abs(Vpgi)) - 1 )
                             * tr - time ) max 0.0
           ; ( delay delayTimeS; search:= false
             | q?
               ; Vpgi:= calcVpgi ( kmF,kmB,keq,Vmax,concS,concP )
               ; side:= cond( Vpgi, concS, concP )
               ; ( side /= +0 -> skip
                 | side  = +0 -> search:= false
                 )
             )
         )
      ; pF!search; pB!search
      ; ( side /= +0 -> procTime:= 1 / Vmax
          ; ( side = +1 -> a?x | side = -1 -> c?x )
          ; delay procTime
          ; ( side = +1 -> b!x | side = -1 -> d!x )
        | side  = +0 -> skip
        )
    )
|| *( rF?concS | rB?concP )
]|
```

Model input:

```
|| Erev(a,b,c,d,rF,rB,q,pF,pB,S0,P0,Vmax,kmF,kmB,keq,tr)
```

D.3 Convertor

D.3.1 Constant rate *C0*

The conversion process `C0` converts molecules with rate `muC`, see Figure D.11.

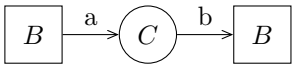

Figure D.11: DEM representation of conversion process *C0*.

Chi file:

```
proc C0( chan a?,b!: mol, val muC: real ) =
|[ var x: mol, at: real
:: *( a?x; at:= 1 / muC; delay at; b!x )
```

```
]|
```

Model input:

```
|| C0( a, b, muC )
```

D.3.2 *ADP* dependent *C1*

The conversion process `C1` converts molecules with rate a rate that depends linearly on the upstream buffer concentration and is maximally `muC`, see Figure D.12.

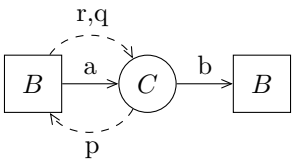

Figure D.12: DEM representation of conversion process *C1*.

Chi file:

```
proc C( chan a?,b!: mol, r?:nat, q?:void, p!:bool
           , val adpmax: nat, muC: real
      ) =
|[ var x: mol
     , adp: nat
     , at: real
     , conv: bool
     , convStart: real
:: *( a?x
      ; r?adp
      ; convStart:= time
      ; at:= 1 / ( muC * (adp + 1) / adpmax )
      ; conv:= true
      ; p!conv
      ; conv
        *> ( delay at; conv:= false
           | q? ; r?adp
             ; at:= (convStart + 1 / (muC * (adp + 1) / adpmax) - time) max 0.0
           )
      ; p!conv
      ; b!x
    )
]|
```

Model input:

```
|| C1( a, b, r, q, p, ADP0 + ATP0, muC )
```

D.4 Data tracker DT

Process `DT` is always ready to receive data from all processes and prints them along with the time the data is received:

```
proc DT( chan dt?: ( string, nat ) ) =
|[ var k: nat, name : string
:: *( dt?( name , k); !! time , "\t", name , "\t", k, "\n" )
]|
```

D.5 Generator G

Process `G` generates substrate molecules with rate `lambdaG` (input variable) and sends them to the downstream buffer:

```
proc G( chan a!: mol, val lambdaG: real ) =
|[ var i: nat = 1, at: real
:: *( a!(i,time); at:= 1 / lambdaG; delay at; i:= i + 1 )
]|
```

D.6 Exit X

Process `X` receives molecules from the product buffer with rate `lambdaE` (input variable).

```
proc X( chan a?: mol, val lambdaE: real ) =
|[ var x: mol, at: real
:: *( a?x; at:= 1 / lambdaE; delay at )
]|
```

Appendix E

Chi models

Every χ model discussed in this report consists of molecule buffers, one or more enzymes and some functions. Since a lot of the processes and functions in the models are similar we present the chi model as a composition of processes and functions. These processes and functions are connected as a parallel composition.

E.1 Substrate enzyme reaction

The model of Chapter 3 is presented below. Figure E.1 shows the processes and communication channels of the substrate-enzyme reaction. The model with processes and functions is presented in Table E.1.

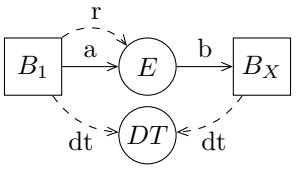

Figure E.1: Discrete event model representation.

Table E.1: Substrate enzyme reaction model.

Chi code	Reference
`type mol = ( nat, real ) // id, timein`	
Process *B1*	(D.1.1)
Process *BX*	(D.1.3)
Process *E*	(D.2.1)
Process *DT*	(D.4)
Function *injBuff*	(C.1)
Function *meanST*	(C.2)

The model is defined by:

```
model L( val S0,P0: nat ) =
|[ chan a,b,g: mol
      , r: nat
      , q: void
      , p: bool
      , dt: (string , nat)
:: DT(dt)
|| B1(g,a,r,q,p,dt,S0,"S")
|| E (a,b,r,q,p, 20.0, 100.1, 0.05)  // value Vmax, Km, tr
|| BX(b,dt,"P")
]|
```

E.2 Substrate enzyme reaction with feedback

The steady-state substrate-enzyme model with feedback from substrate S and product P of Chapter 5 is presented in Table E.2. Figure E.2 shows the processes and communication channels of the steady-state substrate-enzyme system with feedbacks from substrate and product molecule buffers.

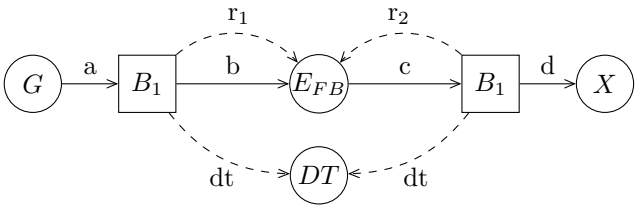

Figure E.2: Discrete event model

Table E.2: Substrate enzyme reaction with feedback model.

Chi code	Reference
`type mol = ( nat, real ) // id, timein`	
Process *G*	(D.5)
Process *B1*	(D.1.1)
Process *Efb*	(D.2.6)
Process *X*	(D.6)
Process *DT*	(D.4)
Function *injBuff*	(C.1)
Function *meanST*	(C.2)

The model is defined by:

```
model L( val S0,P0: nat, parS,parP,lambdaG,lambdaE: real ) =
|[ chan a,b,c,d: mol
      , r,r1 : nat
      , dt: (string , nat)
      , nuS,nuP,q,q1: void
      , tS,tP,p,p1: bool
:: DT(dt)
|| G(a,lambdaG)
|| Bfb1(a,b,r,nuS,q,tS,p,dt,parS,S0,"S")
|| Efb(b,c,r,nuS,nuP,q,tS,tP,p, 20.0, 100.1, 0.05)// value Vmax, Km, tr
|| Bfb0(c,d,nuP,tP,dt,parP,P0,"P")
|| X(d,lambdaE)
]|
```

E.3 Reversible substrate enzyme reaction

The model of Chapter 6 is presented in this section. Figure E.3 shows the processes and communication channels of the steady-state reversible substrate-enzyme reaction. The buffers contain the concentration of respectively the substrate $G6P$ and product $F6P$ molecules. The code is presented in Table E.3.

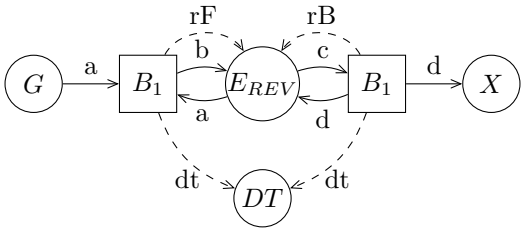

Figure E.3: DEM representation of a steady-state reversible reaction.

Table E.3: Reversible substrate enzyme reaction model.

Chi code	Reference
`type mol = ( nat, real ) // id, timein`	
Process *G*	(D.5)
Process *B1*	(D.1.1)
Process *Erev*	(D.2.8)
Process *X*	(D.6)
Process *DT*	(D.4)
Function *injBuff*	(C.1)
Function *cond*	(C.6)
Function *calcVpgiRev*	(C.5)

The model is defined by:

```
model L( val S0,P0: nat, lambdaG,lambdaE: real ) =
|[ chan a,b,c,d: mol
      , rF,rB: nat
      , dt: (string , nat)
      , q: void
      , pF,pB: bool
:: DT(dt)
|| G(a,lambdaG)
|| B1(a,b,rF,q,pF,dt,S0,"S")
|| Erev(b,c,d,a,rF,rB,q,pF,pB,4647.60935 // value Vmax
     ,3000.0,160.0,0.3,215.16438338347e-6)   // kmF,kmB,keq,tr
|| B1(c,d,rB,q,pB,dt,P0,"P")
|| X(d,lambdaE)
]|
```

E.4 EMP pathway

The model of Chapter 7 is presented here. Figure E.4 shows the processes and communication channels of the EMP pathway. The `dt` channels from each buffer to process `DT` are not shown for complexity of the figure, but are present in the system. This is a steady-state model but can also be used as a transient model if the generator and exit speeds are equal to zero. The conversion process can convert molecules at a given rate or it can convert molecules depending on the ADP concentration. The code is presented in Table E.4.

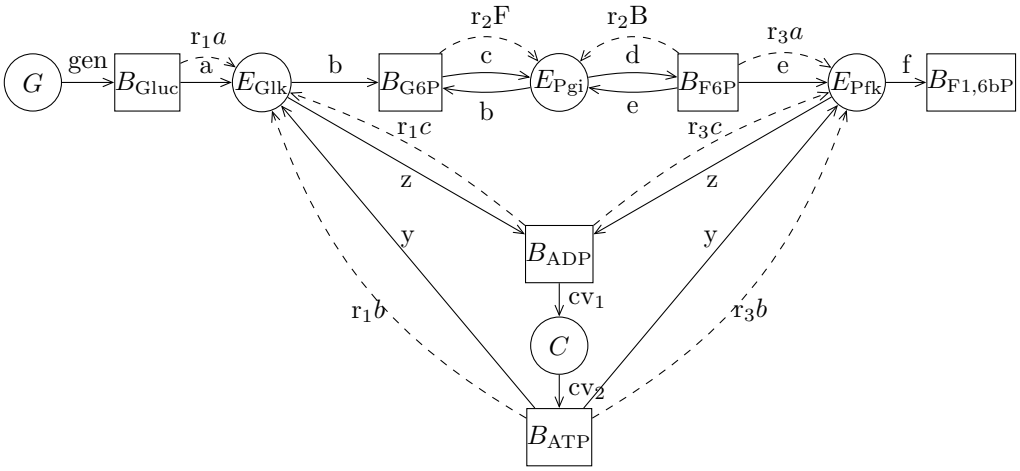

Figure E.4: DEM representation of the EMP-pathway.

Table E.4: EMP pathway model.

Chi code	Reference
`type mol = ( nat, real ) // id, timein`	
Process G	(D.5)
Process *B1*	(D.1.1)
Process *B2*	(D.1.2)
Process *BX*	(D.1.3)
Process *Efb2*	(D.2.7)
Process *Erev*	(D.2.8)
Process X	(D.6)
Process C	(D.3)
Process DT	(D.4)
Function *injBuff*	(C.1)
Function *cond*	(C.6)
Function *calcVpgiRev*	(C.5)
Function *meanST1*	(C.3)

The model is defined by:

```
model L( val c1,c2,c3,c4,c5,c6: nat, lambdaG, muC: real ) =
|[ chan a,b,c,d,e,f,y,z,gen,cv1,cv2: mol
```

```
      , r1a,r1b,r2F,r2B,r3a,r3b: nat
      , q1a,q1b,q2,q3a,q3b: void
      , fS1,fP1,fS2,fP2: void
      , p1a,p1b,p2F,p2B,p3a,p3b: bool
      , tS1,tP1,tS2,tP2: bool
      , dt: (string , nat)
:: DT(dt)
|| G (gen,1/lambdaG)
|| B1 (gen,a,r1a,q1a,p1a,dt,c1,"Gluc") //GLUC
|| Efb2 (a,b,y,z,r1a,r1b,fS1,fP1,q1a,q1b,tS1,tP1,p1a,p1b
        ,442.63982475,120.0,500.0,2259.1731337432e-6,1.0) //e GLK
|| B1 (b,c,r2F,q2,p2F,dt,c2,"G6P") // G6P
|| Erev (c,d,e,b,r2F,r2B,q2,p2F,p2B
       ,4647.60935 ,3000.0,160.0,0.3,215.16438338347e-6) //e PGI
|| B2 (d,e,r2B,r3a,q2,q3a,p2B,p3a,dt,c3,"F6P") // F6P
|| Efb2 (e,f,y,z,r3a,r3b,fS2,fP2,q3a,q3b,tS2,tP2,p3a,p3b
        ,252.5465 ,460.0,40.0,3959.6668336326e-6,1.9 ) //e PFK
|| BX (f,dt,"F16bP") // F16bP
|| B2fb1 (cv2,y,r1b,r3b,fS1,fS2,q1b,q3b,tS1,tS2,p1b,p3b,dt
          ,500.0,2259.173e-6,40.0,3959.67e-6,c5,"ATP") // ATP
|| B2fb0 (z,cv1,fP1,fP2,tP1,tP2,dt
          ,500.0,2259.173e-6,40.0,3959.67e-6,c6,"ADP") // ADP
|| C0 (cv1,cv2,1/muC)
]||
```