2019

Applications of Asymptotic Methods: Analyzing Mathematical Models in Neuroscience and Describing Fast Dynamics of a Trajectory in the Vicinity of a Chaotic Attractor

Denis Mikhailovich Shchepakin

Applications of Asymptotic Methods: Analyzing Mathematical Models in Neuroscience and Describing Fast Dynamics of a Trajectory in the Vicinity of a Chaotic Attractor.

By

Denis Mikhailovich Shchepakin

B.S., Mathematics, Samara State University, Samara, Russia, 2012
M.A., Applied Mathematics, University of Montana, Missoula, Montana, USA, 2014

Dissertation

Presented in partial fulfillment of the requirements
for the degree of

Doctorate of Philosophy
in Mathematics

The University of Montana
Missoula, MT

May 2019

Approved by:

Scott Whittenburg, Dean of The Graduate School
Graduate School

Dr. Leonid Kalachev, Chair
Mathematical Sciences

Dr. Emily F. Stone
Mathematical Sciences

Dr. Brian Steele
Mathematical Sciences

Dr. Johnathan Bardsley
Mathematical Sciences

Dr. Michael P. Kavanaugh
Center for Structural and Functional Neuroscience

Abstract

Shchepakin, Denis, Doctorate of Philosophy, Spring 2019 Mathematics

Applications of Asymptotic Methods: Analyzing Mathematical Models in Neuroscience and Describing Fast Dynamics of a Trajectory in the Vicinity of a Chaotic Attractor.

Committee Chair: Leonid Kalachev, Ph.D.

The current dissertation focuses on two unrelated subjects: modeling in Neuroscience applications and Chaos Theory.

Neurons are units of the nervous system that receive, conduct, and transmit information to each other and target tissue via electrical signaling. One of the mechanisms of the signal transduction is through signaling molecules called neurotransmitters. Glutamate is the predominant excitatory neurotransmitter in the mammalian and human central nervous system. However, the mechanism of regulation and sensation of the glutamate via glutamate receptors and transporters is not completely understood.

We discuss currently existing models of glutamate receptors and transporters, and their main problem: the overparameterization with respect to the existing experimental data. Although this issue prevents statistical reliable parameter estimate, numerous authors still attempt to use them for these means using incorrect methodology. We are able to reduce the existing models under certain assumptions, that are achieved in experiments, designed specifically for this purpose. This approach allows us to avoid the overparameterization issue and for the first time obtained statistically reliable parameter estimates.

Chaotic systems do not allow for conventional methods of parameter estimation and had been considered to be an ill–posed problem. Recently a novel promising methodology was proposed. Here we discuss some further development of the technique that brings it closer to practical use.

Contents

1 Introduction

The current dissertation focuses on two unrelated subjects: modeling of glutamate receptor and transporters as well as synaptic transmission and extension of a novel parameter estimation approach in Chaos Theory. Additional background information and context are given in the corresponding sections. Let us outline them.

In Section 2 we introduce some basic concepts from the Neuroscience field. Section 3 begins with background information about glutamate transporters, overview of the existing models of glutamate transporters, and issues related to estimating transporters' turnover rates. The section continues with a derivation and application of a new model that resolves mentioned issues. The real experimental data are used to obtain statistically reliable turnover rates estimates. The section is then continues the discussion of the role of glutamate transporters in clearing synaptic cleft during synaptic transmission. The 3–dimensional diffusion model of a synapse is constructed using the turnover rates estimates obtained on the previous step. The diffusion model is then analyzed in a context of a synapse functioning. Section 4 focuses on glutamate receptors and discusses a specific phenomenon called desensitization. The section goes through the difficulties of identifying the nature of the phenomenon. A series of receptor's models and experiments' designs are then constructed in conjunction in order to overcome the stated difficulties. The section finishes with a proposed algorithm of experiments and fitting procedures that yields some insight on the nature of receptor's desensitization.

Section 5 presents a recently developed novel method of estimating parameters of chaotic systems. A useful extension of the methodology is discussed.

Appendix A contains useful results from the literature that are used here. Appendix B contains MATLAB code files.

The results from Sections 3 — 5 are published here:

1. Denis Shchepakin, Leonid Kalachev, and Michael Kavanaugh, *Modeling of Excitatory Amino Acid Transporters*, Research Perspectives CRM Barcelona, Summer 2018, vol. 11, in Trends in Mathematics, Springer-Birkhäuser, Basel (in press).

2. Denis Shchepakin, Leonid Kalachev, and Michael Kavanaugh, *Modeling of N–methyl–D–aspartate receptors*, Research Perspectives CRM Barcelona, Summer 2018, vol. 11, in Trends in Mathematics,Springer-Birkhäuser, Basel (in press).

3. Denis Shchepakin, Leonid Kalachev, and Michael Kavanaugh, *Modeling of excitatory amino acid transporters and clearance of synaptic cleft on millisecond time scale*, Mathematical Modelling of Natural Phenomena (in press).

4. Sebastian Springer, Heikki Haario, Leonid Kalachev, Vladimir Shemyakin, Denis Shchepakin, *Robust Parameter Estimation of Chaotic System*, Inverse Problems and Imaging (in press).

2 Neuroscience

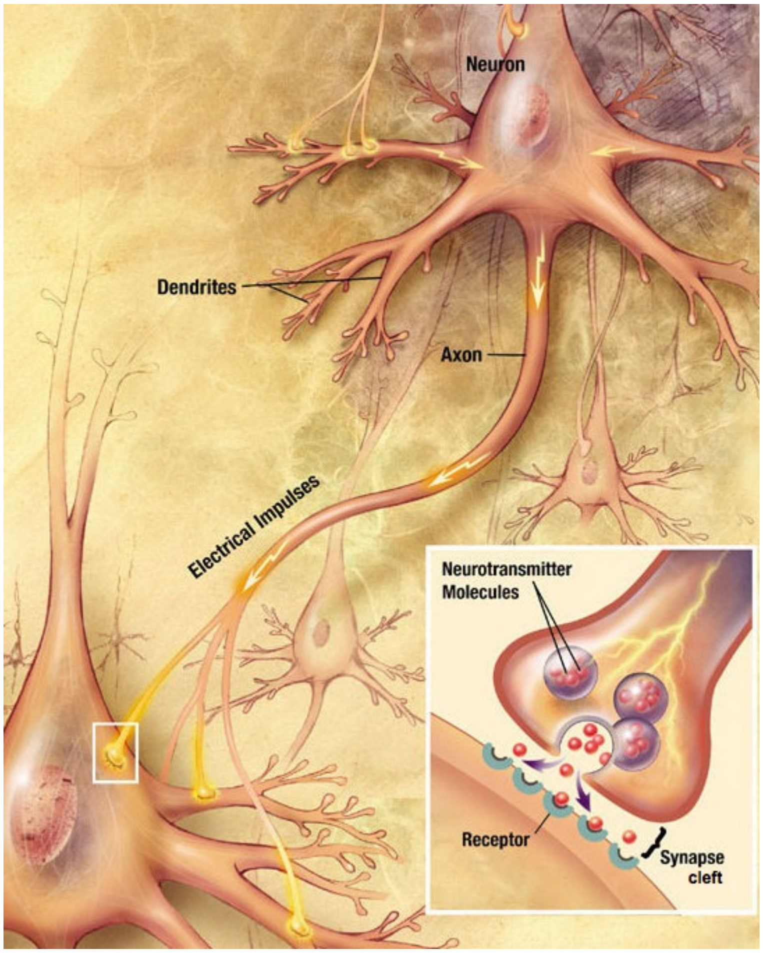

Figure 1: Neuron and synapse structures (http://commons.wikimedia.org/wiki/File:Chemical_synapse_schema_cropped.jpg)

Typical neuron consists of a cell body and multiple extensions called neurites. There are two types of neurites: dendrites and axons. Dendrites are small highly branched processes, usually a fraction of a millimeter in length, and serve as neuron's signal receivers from other neurons. There is typically a single axon in a neuron, which is longer than a dendrite (e.g., up to about 1 meter in humans). An axon transmits a signal from a neuron to other neurons or target tissue. As a neuron receives a signal through its dendrites, membrane-embedded proteins, ion-channels, activate and change its cross-membrane potential. If the accumulated change reaches a certain threshold, the neuron will generate a one-way electrochemical pulse along the axon, called action potential, which transmits signal down the neural circuit. When an action potential reaches the end of the axon, an axon terminal, the signal must be transmitted through a junction, called a synapse, to the next cell. There are two types of synapses: electrical and chemical. In this work we focus on chemical synapses. A chemical synapse is a small gap between an axon terminal of a signaling neuron, called pre-synaptic cell, and a dendrite of a receiving neuron or other target cell, called post-synaptic cell. Upon receiving an action potential, the pre-synaptic cell releases special signaling molecules, called neurotransmitters, that are stored within the cell in the synaptic vesicles. The neurotransmitter diffuses through the synaptic cleft and is sensed by the post-synaptic cell via specialized receptors, which in turn change the post-synaptic cell potential via various mechanisms; see Figure 1. Depending on the type of neurotransmitter and the receptors, this change could be excitatory, i.e., bringing the potential of a cell closer to the action potential threshold, or inhibitory, i.e., producing the opposite effect. After a short amount of time (few milliseconds), the neurotransmitter leaves the synaptic cleft through a process of degradation or physical uptake by other specialized proteins, called transporters. This is necessary for information transfer efficacy. Moreover, prolonged exposure of a post-synaptic cell to some neurotransmitter, e.g., glutamate, could lead to nerve damage or death, the effect which is called excitotoxicity. Therefore, the termination of the neurotransmitter signaling also plays a protective role. Here we focus on a particular signaling molecule, glutamate. It is the most abundant excitatory neurotransmitter in mammalian central nervous system (CNS). The disruption of the glutamate regulation in the brain could lead to such problems as amyotrophic lateral sclerosis, Alzheimer's disease and neuronal death [1, 2].

3 Glutamate transporters

There are no reported extracellular enzymes that degrade glutamate, thus, the role of glutamate transporters is of high importance. The glutamate transporters that mainly accomplish this task are excitatory amino acid transporters (EAATs). To date, five major subtypes have been identified (EAAT1—5) in mammalian CNS, that are all part of solute carrier family 1 (SLC1) [3, 4]. EAATs are found on the membranes of the neurons and astrocytes, a type of glial cells that play various roles in central nervous system, including a maintenance of extracellular ion balance. Astrocytes envelop some of the synapses to various degrees [5] and their transporters play a role in the clearance of a glutamate from the synaptic cleft. EAAT3 are exclusively expressed on neurons' dendrites and somas and EAAT1 are found on astrocytes. EAAT2 are expressed on both astrocytes and synapses of presynaptic cells [6, 7]. EAAT4 is found on the dendrites of the cerebellar Purkinje cells and in some neurons of forebrain and EAAT5. EAAT4—5 are not very efficient transporters and were suggested to play a role analogous to an inhibitory glutamate receptors [3]. Therefore, here we focus on EAAT1—3.

Although our understanding of EAATs' functioning is crucial for various reasons, e.g., modeling or drug developing, even their most basic characteristics are poorly estimated. The reports of the transporters' turnover rate, i.e., the average number of glutamate molecules that can be transferred from the extracellular space across the membrane by a single transporter per unit of time, ranges in the literature from a few molecules per second on the lower end [8, 9] to the numbers that are several fold greater [10, 11]. Even the largest estimated values are few orders of magnitude slower than the synaptic transmission [21, 22]. Nevertheless, the glutamate transporters are reported to somehow influence the dynamics of a much faster synaptic signaling [23, 24, 25, 26, 27].

Here we derive a model for a specific experiment design, which allows for a reliable estimate of EAAT1—3 turnover rates. Next, we construct a 3—dimensional model of a synapse and using the obtained turnover rates estimates simulate the synaptic transmission. This allows us to get the insight on the nature of discrepancies in both EAATs turnover rates and their ability to influence fast synaptic signaling.

3.1 Turnover rate of the glutamate transporters

3.1.1 Current models

We begin with the basic knowledge of how EAATs transport glutamate across the membrane and discuss the corresponding widely-accepted straight-forward chemical kinetics model and its limitation. EAATs transport glutamate molecules against their electrochemical gradient via the electrochemical gradients of other ions: with each molecule of glutamate 3 Na^+ and 1 H^+ are co-transported, and 1 K^+ is counter-transported [12, 13]. Therefore, the net charge of +2 moves into the cell with each full cycle of the transporter. Electrophysiological experimental techniques allow to record this movement of charges in cells expressing the transporters and relate the data to glutamate uptake. The order of ion binding to and unbinding from the transporter has been established and reflected in the commonly accepted chemical kinetics model (see Figure 2) [14, 15]. Glutamate transporters also play a role of a chloride channel [9, 16], i.e., Cl^- can be transferred through some of the states of the model resulting in an additional charge movement. The exact states that can interact in such a way with chloride ions are not known. If we assume that this is possible for all states, this will increase the number of states in the model to twice as many.

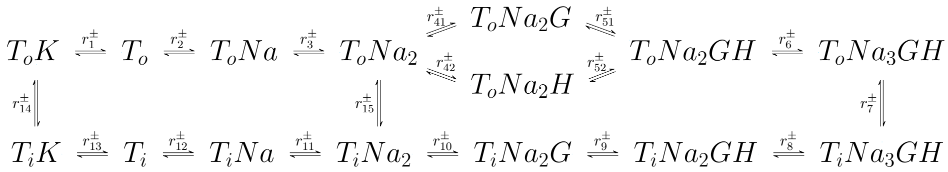

Figure 2: 15-state model for EAATs. T_o and T_i stand for transporter facing extracellular and intracellular spaces, respectively; Na, K, H denote corresponding ions; G is L–glutamate; $r_i^\pm$ for all i are reaction rate constants. r_i^+ and r_i^- correspond to a clockwise and counterclockwise directions, respectively.

The practical use of the described 15—state model (30—state with Cl^- conduction) is highly limited as the number of unknown parameters, i.e., reaction rates, is beyond that which can be estimated from the experimental data, that is, the model is overparameterized with respect to the existing data. One of the possible solutions to resolve the issue is to reduce the model when some specific conditions can be maintained, i.e., a controlled experiment. We consider a certain technique, called an excised patch clamp experiment. Let us provide a short overview of the procedure. An investigator forms a tight seal between a cell membrane expressing the transporters and a fine micropipette. The micropipette is then withdrawn from the cell, which causes the cell membrane to bulge and eventually rupture, resealing its small portion, the "patch", on the pipette. The obtained patch retains the local structure of the cell membrane, including transmembrane proteins like ion channels and transporters. Depending on the technique of pulling, the patch can face either of two ways: the cytosolic (inner) surface of the patch faces the inside of the pipette (called "outside—out patch") or the outside of the pipette (called "inside—out patch"). Content of both solutions (inside and outside of the pipette) is controlled by the experimenter and simulate intracellular and extracellular environment of the cell for the patch. The solution in the bath, i.e., outside of the micropipette, can be rapidly switched by physically moving the pipette or the bath. A piezo electric switch is often used for such purposes and it allows for a solution change within a few milliseconds. The electrodes inside the pipette and in the bath are then used to study the flow of ions across the membrane, i.e., patch, that should emulate the flow of ions across the membrane of the whole cell but on a smaller scales, which allows for a tight control of the solution on both sides of the cell wall.

Let us discuss how this experimental approach can be used to reduce the chemical kinetics model. Since one has the total control of the solutions that represent intracellular and extracellular space in excised patch clamp experiments, the ion concentrations can be adjusted in such a way that they accelerate the transitions of the model states in a clockwise manner. The increase of sodium and hydrogen ions, and glutamate concentrations in the solution representing extracellular space will accelerate their binding to the transporter; the reduction of K^+ concentration will facilitate its unbinding from the transporter ($T_oK \to T_o$ transition). The analogous approach can be taken for the solution representing intracellular environment as well. This approach will saturate the transporter with its ligands and increase the probability of transition in the clock-wise manner (see Figure 2 for visual reference), making some of these transitions almost instantaneous, which will effectively eliminate some of the model states. Under the described conditions the number of effective states in the model decreases to four. After addition of four more states that are responsible for conductance of chloride ions, the reduced model is obtained; see Figure 3. Let us note that similar models have been proposed and studied before, see e.g., [9, 10, 17, 18]. Although these reduced models seem to retain many, if not all, of the qualitative features of recorded currents, they still cannot be used for parameter estimation as they remain to be overparameterized with respect to the experimental data. Thus, further reduction is needed.

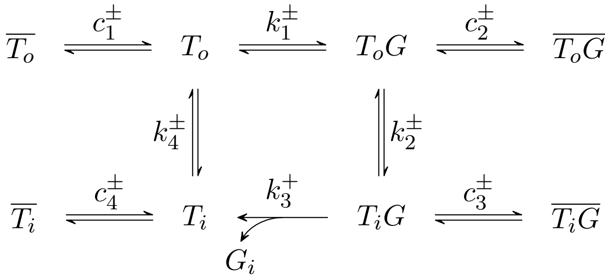

Figure 3: The simplified 8-state model for the patch clamp experiment. The states with bars are the corresponding conducting states, which allow the flow of chloride ions.

3.1.2 Model derivation

Now, let us discuss the possible further reduction of the 8–state model (Figure 3). Its chemical kinetics scheme can be represented by a system of differential equations using the rate law. For each state we define a corresponding variable, using the same notation, which is a fraction of all transporters observed to be in this state (e.g., the variable $T_o(t)$

is a fraction of all transporters in T_o state). The system has the following form:

$$\begin{aligned}
\frac{dT_o}{dt} &= c_1^+\overline{T_o} - c_1^- T_o - (k_1^+ + k_4^-)T_o + k_1^- T_oG + k_4^+ T_i,\\
\frac{dT_oG}{dt} &= -c_2^+ T_oG + c_2^- \overline{T_oG} - (k_1^- + k_2^+)T_oG + k_1^+ T_o + k_2^- T_iG,\\
\frac{dT_iG}{dt} &= -c_3^+ T_iG + c_3^- \overline{T_iG} - (k_2^- + k_3^+)T_iG + k_2^+ T_oG,\\
\frac{dT_i}{dt} &= c_4^+\overline{T_i} - c_4^- T_i - k_4^+ T_i + k_3^+ T_iG + k_4^- T_o,\\
\frac{d\overline{T_o}}{dt} &= -c_1^+\overline{T_o} + c_1^- T_o,\\
\frac{d\overline{T_oG}}{dt} &= c_2^+ T_oG - c_2^- \overline{T_oG},\\
\frac{d\overline{T_iG}}{dt} &= c_3^+ T_iG - c_3^- \overline{T_iG},\\
\frac{d\overline{T_i}}{dt} &= -c_4^+\overline{T_i} + c_4^- T_i.
\end{aligned} \tag{1}$$

We focus on the reduction for the outside—out patch experiment (so the 8–state model derived under these conditions is valid), i.e., the bath solution represents the extracellular environment. During the experiments we switch back and forth between two bath mixtures with the same substances content except for glutamate: one solution contains a saturating concentration of the neurotransmitter and another does not have any at all. Thus, each switch of the solution makes an abrupt change in an "extracellular" glutamate concentration. When glutamate is not present in the bath, a transition of the transporter from the T_o state to T_oG is impossible, without directly affecting other transition. This can be modeled by simply putting reaction rate k_1^+ to zero. Because k_1^+ value effectively switches back and forth between zero and non-zero values every time we switch the bath solution, we have split the modeling of the experiment into the corresponding stages. Naturally, we will use the end state of the system from the previous stage as the initial conditions for the system in the next stage.

As we mentioned, the use of model (1) for estimating model parameters is not a valid approach due to overparameterization of the system with respect to the existing experimental data (and, thus, the need for the reduction). According to [9], the transitions to conducting states are much faster compared to the reactions which correspond to glutamate transportation, i.e. $c_i^\pm \gg k_j^\pm$ for all i, j. We can define a small parameter $0 < \varepsilon \ll 1$ in the following way:

$$c_i^\pm = \frac{\widetilde{c_i^\pm}}{\varepsilon}, \quad \widetilde{c_i^\pm} \sim O(1), \quad k_i^\pm \sim O(1), \quad i = 1, 2, 3, 4. \tag{2}$$

The boundary function method (Appendix A.1) allows for the reduction of system (1). We represent all the variables as asymptotic series with respect to the small parameter ε. We note that we will use slightly different notations for the regular functions to avoid double subscripts:

$$\begin{array}{rclclc}
T_o(t) & = & T_o^{(0)}(t) + \Pi_0 T_o(\tau) & + & \varepsilon\left(T_o^{(1)}(t) + \Pi_1 T_o(\tau)\right) & + \quad \cdots, \\
T_oG(t) & = & T_oG^{(0)}(t) + \Pi_0 T_oG(\tau) & + & \varepsilon\left(T_oG^{(1)}(t) + \Pi_1 T_oG(\tau)\right) & + \quad \cdots, \\
T_iG(t) & = & T_iG^{(0)}(t) + \Pi_0 T_iG(\tau) & + & \varepsilon\left(T_iG^{(1)}(t) + \Pi_1 T_iG(\tau)\right) & + \quad \cdots, \\
 & & & \vdots & & \\
\overline{T_i}(t) & = & \overline{T_i}^{(0)}(t) + \Pi_0 \overline{T_i}(\tau) & + & \varepsilon\left(\overline{T_i}^{(1)}(t) + \Pi_1 \overline{T_i}(\tau)\right) & + \quad \cdots,
\end{array} \tag{3}$$

where $\tau = t/\varepsilon$ is a rescaled (stretched) time variable. Let us remind that the terms in the asymptotic series that depend on t are called the regular functions, and the terms that depend on τ are called the boundary functions. Next, we substitute (2) and (3) into the system (1). The resulting system is quite big, so we demonstrate only one equation of the resulting system as an example to give the idea of how the methodology works, with all other equations being similar. For example, the first equation of the system (1) becomes:

$$\begin{aligned}
\frac{dT_o^{(0)}}{dt} + \frac{1}{\varepsilon}\frac{d\Pi_0 T_o}{d\tau} + \varepsilon\frac{dT_o^{(1)}}{dt} + \frac{d\Pi_0 T_o}{d\tau} + \cdots \\
= \frac{\widetilde{c_1^+}}{\varepsilon}\left(\overline{T_o}^{(0)}(t) + \Pi_0\overline{T_o}(\tau) + \varepsilon\overline{T_o}^{(1)}(t) + \varepsilon\Pi_1\overline{T_o}(\tau) + \cdots\right) \\
-\frac{\widetilde{c_1^-}}{\varepsilon}\left(T_o^{(0)}(t) + \Pi_0 T_o(\tau) + \varepsilon T_o^{(1)}(t) + \varepsilon\Pi_1 T_o(\tau) + \cdots\right) \\
-(k_1^+ + k_4^-)\left(T_o^{(0)}(t) + \Pi_0 T_o(\tau) + \varepsilon T_o^{(1)}(t) + \varepsilon\Pi_1 T_o(\tau) + \cdots\right) \\
+k_1^-\left(T_oG^{(0)}(t) + \Pi_0 T_oG(\tau) + \varepsilon T_oG^{(1)}(t) + \varepsilon\Pi_1 T_oG(\tau) + \cdots\right) \\
+k_4^+\left(T_i^{(0)}(t) + \Pi_0 T_i(\tau) + \varepsilon T_i^{(1)}(t) + \varepsilon\Pi_1 T_i(\tau) + \cdots\right).
\end{aligned}$$

Equating coefficients of like powers of ε, separately for regular and boundary layer functions, we obtain the reduced model problems for different terms of the asymptotic expansion (3). For regular functions in the leading order approximation, we obtain the following system:

$$
\begin{aligned}
\overline{T_o}^{(0)} &= \frac{\widetilde{c_1^-}}{\widetilde{c_1^+}} T_o^{(0)} = \frac{c_1^-}{c_1^+} T_o^{(0)}, \\
\overline{T_oG}^{(0)} &= \frac{c_2^+}{c_2^-} T_oG^{(0)}, \\
\overline{T_iG}^{(0)} &= \frac{c_3^+}{c_3^-} T_iG^{(0)}, \\
\overline{T_i}^{(0)} &= \frac{c_4^-}{c_4^+} T_i^{(0)}, \\
\frac{dT_o^{(0)}}{dt} &= \frac{c_1^+}{c_1^+ + c_1^-} \left(-(k_1^+ + k_4^-) T_o^{(0)} + k_1^- T_oG^{(0)} + k_4^+ T_i^{(0)} \right), \\
\frac{dT_oG^{(0)}}{dt} &= \frac{c_2^-}{c_2^+ + c_2^-} \left(-(k_1^- + k_2^+) T_oG^{(0)} + k_1^+ T_o^{(0)} + k_2^- T_iG^{(0)} \right), \\
\frac{dT_iG^{(0)}}{dt} &= \frac{c_3^-}{c_3^+ + c_3^-} \left(-(k_2^- + k_3^+) T_iG^{(0)} + k_2^+ T_oG^{(0)} \right), \\
\frac{dT_i^{(0)}}{dt} &= \frac{c_4^+}{c_4^+ + c_4^-} \left(-k_4^+ T_i^{(0)} + k_3^+ T_iG^{(0)} + k_4^- T_o^{(0)} \right).
\end{aligned}
\tag{4}
$$

Let us introduce the following notations:

$$
\begin{aligned}
m_1^+ &= \frac{c_1^+ k_1^+}{c_1^+ + c_1^-}, \qquad & m_1^- &= \frac{c_2^- k_1^-}{c_2^+ + c_2^-}, \\
m_2^+ &= \frac{c_2^- k_2^+}{c_2^+ + c_2^-}, \qquad & m_2^- &= \frac{c_3^- k_2^-}{c_3^+ + c_3^-}, \\
m_3^+ &= \frac{c_3^- k_3^+}{c_3^+ + c_3^-}, & & \\
m_4^+ &= \frac{c_4^+ k_4^+}{c_4^+ + c_4^-}, \qquad & m_4^- &= \frac{c_1^+ k_4^-}{c_1^+ + c_1^-},
\end{aligned}
$$

and

$$
\begin{aligned}
x &= \frac{c_1^+ + c_1^-}{c_1^+} T_o^{(0)}, \qquad & y &= \frac{c_2^+ + c_2^-}{c_2^-} T_oG^{(0)}, \\
z &= \frac{c_3^+ + c_3^-}{c_3^-} T_iG^{(0)}, \qquad & w &= \frac{c_4^+ + c_4^-}{c_4^+} T_i^{(0)}.
\end{aligned}
$$

Then, by solving the following system

$$\begin{aligned}
\frac{dx}{dt} &= -m_1^+ x + m_1^- y + m_4^+ w - m_4^- x,\\
\frac{dy}{dt} &= m_1^+ x - m_1^- y - m_2^+ y + m_2^- z,\\
\frac{dz}{dt} &= m_2^+ y - m_2^- z - m_3^+ z,\\
\frac{dw}{dt} &= m_3^+ z - m_4^+ w + m_4^- x,
\end{aligned} \tag{5}$$

we can find the regular functions in the leading order approximation:

$$\begin{aligned}
T_o(t) \ &= \frac{c_1^+}{c_1^+ + c_1^-} x(t) + O(\varepsilon), & \overline{T_o}(t) \ &= \frac{c_1^-}{c_1^+ + c_1^-} x(t) + O(\varepsilon),\\
T_oG(t) &= \frac{c_2^-}{c_2^- + c_2^+} y(t) + O(\varepsilon), & \overline{T_oG}(t) &= \frac{c_2^+}{c_2^- + c_2^+} y(t) + O(\varepsilon),\\
T_iG(t) &= \frac{c_3^-}{c_3^- + c_3^+} z(t) + O(\varepsilon), & \overline{T_iG}(t) &= \frac{c_3^+}{c_3^- + c_3^+} z(t) + O(\varepsilon),\\
T_i(t) \ &= \frac{c_4^+}{c_4^+ + c_4^-} w(t) + O(\varepsilon), & \overline{T_i}(t) \ &= \frac{c_4^-}{c_4^+ + c_4^-} w(t) + O(\varepsilon).
\end{aligned} \tag{6}$$

For the boundary functions in the leading order approximation, we obtain the following system:

$$\begin{aligned}
\frac{d\Pi_0 T_o}{d\tau} \ &= \widetilde{c_1^+}\Pi_0\overline{T_o} - \widetilde{c_1^-}\Pi_0 T_o,\\
\frac{d\Pi_0 T_oG}{d\tau} &= -\widetilde{c_2^+}\Pi_0 T_oG + \widetilde{c_2^-}\Pi_0\overline{T_oG},\\
\frac{d\Pi_0 T_iG}{d\tau} &= -\widetilde{c_3^+}\Pi_0 T_iG + \widetilde{c_3^-}\Pi_0\overline{T_iG},\\
\frac{d\Pi_0 T_i}{d\tau} \ &= \widetilde{c_4^+}\Pi_0\overline{T_i} - \widetilde{c_4^-}\Pi_0 T_i,\\
\frac{d\Pi_0 \overline{T_o}}{d\tau} \ &= -\widetilde{c_1^+}\Pi_0\overline{T_o} + \widetilde{c_1^-}\Pi_0 T_o,\\
\frac{d\Pi_0 \overline{T_oG}}{d\tau} &= \widetilde{c_2^+}\Pi_0 T_oG - \widetilde{c_2^-}\Pi_0\overline{T_oG},\\
\frac{d\Pi_0 \overline{T_iG}}{d\tau} &= \widetilde{c_3^+}\Pi_0 T_iG - \widetilde{c_3^-}\Pi_0\overline{T_iG},\\
\frac{d\Pi_0 \overline{T_i}}{d\tau} \ &= -\widetilde{c_4^+}\Pi_0\overline{T_i} + \widetilde{c_4^-}\Pi_0 T_i.
\end{aligned}$$

Let us notice that the sum of derivatives of boundary functions of a conducting and a corresponding non-conducting states is zero for every state. Therefore, the sum of boundary functions of a conducting and a corresponding non-conducting state is always a constant. Since all the boundary functions must tend to 0 as τ goes to infinity, all these constants must equal to zero. We have

$$\begin{aligned}
\Pi_0 T_o(\tau) &= \Pi_0 T_o(0) e^{-\left(\widetilde{c_1^+}+\widetilde{c_1^-}\right)\tau}, \\
\Pi_0 T_o G(\tau) &= \Pi_0 T_o G(0) e^{-\left(\widetilde{c_2^+}+\widetilde{c_2^-}\right)\tau}, \\
\Pi_0 T_i G(\tau) &= \Pi_0 T_i G(0) e^{-\left(\widetilde{c_3^+}+\widetilde{c_3^-}\right)\tau}, \\
\Pi_0 T_i(\tau) &= \Pi_0 T_i(0) e^{-\left(\widetilde{c_4^+}+\widetilde{c_4^-}\right)\tau}, \\
\Pi_0 \overline{T_o}(\tau) &= -\Pi_0 T_o(0) e^{-\left(\widetilde{c_1^+}+\widetilde{c_1^-}\right)\tau}, \\
\Pi_0 \overline{T_o G}(\tau) &= -\Pi_0 T_o G(0) e^{-\left(\widetilde{c_2^+}+\widetilde{c_2^-}\right)\tau}, \\
\Pi_0 \overline{T_i G}(\tau) &= -\Pi_0 T_i G(0) e^{-\left(\widetilde{c_3^+}+\widetilde{c_3^-}\right)\tau}, \\
\Pi_0 \overline{T_i}(\tau) &= -\Pi_0 T_i(0) e^{-\left(\widetilde{c_4^+}+\widetilde{c_4^-}\right)\tau}.
\end{aligned} \tag{7}$$

Let us now consider the initial conditions for the states T_o and $\overline{T_o}$, and their leading order approximations:

$$\begin{aligned}
T_o^{(0)}(0) + \Pi_0 T_o(0) &= T_o(0), \\
\overline{T_o}^{(0)}(0) + \Pi_0 \overline{T_o}(0) &= \overline{T_o}(0).
\end{aligned}$$

Using the first equation of (4) and $\Pi_0 T_o = -\Pi_0 \overline{T_o}$ from (7), we get

$$\begin{aligned}
T_o^{(0)}(0) + \Pi_0 T_o(0) &= T_o(0), \\
\frac{c_1^-}{c_1^+} T_o^{(0)}(0) - \Pi_0 T_o(0) &= \overline{T_o}(0).
\end{aligned}$$

We can sum the equations and get an initial conditions for the function $x(t)$ of system (5). Repeating the same procedure for the functions $y(t)$, $z(t)$, and $w(t)$, we obtain:

$$\begin{aligned}
x(0) &= T_o(0) &&+ \overline{T_o}(0), \\
y(0) &= T_o G(0) &&+ \overline{T_o G}(0), \\
z(0) &= T_i G(0) &&+ \overline{T_i G}(0), \\
w(0) &= T_i(0) &&+ \overline{T_i}(0).
\end{aligned}$$

As mentioned earlier, all the experiments start without glutamate in the bath. Thus, the initial conditions before the first switch correspond to the steady–state of the system (1) in the absence of glutamate ($k_1^+ = 0$). In terms of the new notations they become

$$\begin{aligned} x(0) &= \frac{m_4^+}{m_4^+ + m_4^-}, \\ y(0) &= 0, \\ z(0) &= 0, \\ w(0) &= \frac{m_4^-}{m_4^- + m_4^+}. \end{aligned} \tag{8}$$

Moreover, let us notice, while in this steady–state, i.e., at the start of all experiments, the following relations are true for the original states:

$$\begin{aligned} \overline{T_o} &= \frac{c_1^-}{c_1^+} T_o, \\ \overline{T_o G} &= \frac{c_2^+}{c_2^-} T_o G, \\ \overline{T_i G} &= \frac{c_3^+}{c_3^-} T_i G, \\ \overline{T_i} &= \frac{c_4^-}{c_4^+} T_i. \end{aligned}$$

This could be easily seen if one considers last four equations of the system (1) and sets derivatives to zeros. Next, from the first four equations of the system (4) we notice that the regular functions of the leading order approximation have exactly the same relationship at any given time, including at the start of an experiment. From (3) it follows that the boundary functions of the leading order approximation must have the same relations as well:

$$\begin{aligned} \overline{\Pi_0 T_o} &= \frac{c_1^-}{c_1^+} \Pi_0 T_o, \\ \overline{\Pi_0 T_o G} &= \frac{c_2^+}{c_2^-} \Pi_0 T_o G, \\ \overline{\Pi_0 T_i G} &= \frac{c_3^+}{c_3^-} \Pi_0 T_i G, \\ \overline{\Pi_0 T_i} &= \frac{c_4^-}{c_4^+} \Pi_0 T_i. \end{aligned}$$

We have already discussed that the sum of boundary functions of a conducting and a corresponding non-conducting state is always zero, which one can also see from (7). To satisfy both conditions the boundary functions of the leading order approximation must be equal to zero. Every time the switch of the bath solution happens, the boundary functions may potentially appear. However, similar logic forces zero boundary functions every time. Right before the switch, the system is approximated by a leading order approximation with zero boundary functions and, according to first four equations of the system (4), obeys the same relations. At the moment of the switch, the end state of the system becomes the initial conditions for the new system, which is the same system (4) with switched value of k_1^+ and, therefore, must follow the same relations. As before, the boundary conditions must satisfy these relations and (7), which is possible only when they are equal to zero. Therefore, relations (6) describe leading order approximations (both regular and boundary functions) and we only need to solve system (5) to find them.

Let us note that the sum of the variables entering system (5) is constant and it equals to the sum of all the variables in system (1), which is 1. Moreover, the sum of all right hand sides is zero, which means that it is possible to immediately draw a chemical kinetics scheme that corresponds to the system (5) in the same manner as the scheme depicted in Figure 3 relates to (1); see Figure 4.

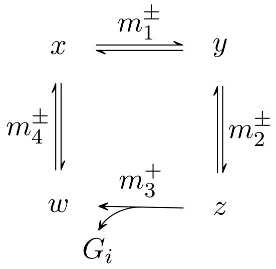

Figure 4: The chemical kinetics scheme that corresponds to the reduced model (5).

The electrophysiology technique allows us to record the current that corresponds to the flow of ions. Therefore, we cannot observe any of the variables of the system (5) directly but rather some sort of their combination. Let us derive the formula for the current recorded during the experiment. The total recorded current is a sum of the stoichiometric current (coupled flux of glutamate molecules and ions across the membrane), the conductive current due to the flow of chloride ions, and some constant leak current [9, 17]. The previously described flow of ions suggests the movement of +3 charge inside the cell with the $T_oG \to T_iG$ transition ($3Na^+$ and H^-) and the movement of +1 charge outside of the cell (i.e., move of −1 charge inside the cell) with the $T_i \to T_o$ transition (K^+). However, the voltage—dependence of the transport suggests that the transporter also mediates a capacitive charge transfer, i.e., it has negative one charge on its own. Thus, the charge of +2 moves during the transition from state T_oG to T_iG with all other transitions having no immediate effect on the transportation current. The chloride-related current is proportional to the fractions of transporters observed in the conductive states ($\overline{T_o}$, $\overline{T_oG}$, $\overline{T_iG}$, and $\overline{T_i}$). The resulting formula for the total current is:

$$I = -A \cdot \overline{T_o} - B \cdot \overline{T_oG} - C \cdot \overline{T_iG} - D \cdot \overline{T_i} - E\left(k_2^+ T_oG - k_2^- T_iG\right) + I_{leak}.$$

Using (6), we can write the leading order approximation for the current I_0 that depends on x, y, z, and w. Since the system is in the steady–state before the glutamate is introduced for the first time, i.e., in the state (8), we can subtract the steady–state current in the above formula in order to identify the leading order approximation of the constant value of I_{leak}. In order to further reduce the number of parameters, we can express any one of the four variables in terms of the others since $x(t) + y(t) + z(t) + w(t) = 1$ for any given instant of time t. We obtain the following formula:

$$I_0(t) = -\mathcal{A} \cdot x(t) - \mathcal{B} \cdot y(t) - \mathcal{D} \cdot w(t) + \frac{\mathcal{A} m_4^+ + \mathcal{D} m_4^-}{m_4^+ + m_4^-} + O(\varepsilon), \tag{9}$$

where

$$\begin{aligned}
\mathcal{A} &= \frac{c_1^- A}{c_1^+ + c_1^-} + \frac{c_3^+ C - c_3^- k_2^- E}{c_3^- + c_3^+}, \\
\mathcal{B} &= \frac{c_2^+ B + c_2^- k_2^+ E}{c_2^- + c_2^+} + \frac{c_3^+ C - c_3^- k_2^- E}{c_3^- + c_3^+}, \\
\mathcal{D} &= \frac{c_4^- D}{c_4^+ + c_4^-} + \frac{c_3^+ C - c_3^- k_2^- E}{c_3^- + c_3^+}.
\end{aligned}$$

Finally, the turnover rate of a particular transporter is equal to the influx of the glutamate molecules through the transportation cycle in the steady–state. Note, that although the transporter' states can reach steady–state in the presence of glutamate, this is a dynamic equilibrium and the glutamate uptake persists, i.e., glutamate is continuously accumulated inside the pipette. That means that $\frac{dG_i}{dt}$ is not zero in a steady–state, but reaches some constant value that we denote as Ξ. It is also worth mentioning that the continuous accumulation of glutamate will lead to the change of kinetics and other changes in the real cell, but, as we stated earlier, the duration of the experiment and the amount of particles transferred during the experiment are too low for such thing to happen and have any detrimental effect on the patch. The turnover rate can is given by

$$\Xi = \lim_{t\to\infty} \frac{dG_i}{dt} = \lim_{t\to\infty} k_3^+ T_i G(t) = k_3^+ \lim_{t\to\infty} T_i G(t) =$$
$$\frac{k_3^+}{\frac{c_4^+ + c_4^-}{c_4^+} \frac{k_3^+}{k_4^+} + \frac{c_3^+ + c_3^-}{c_3^-} + \frac{c_2^+ + c_2^-}{c_2^-} \frac{k_3^+ + k_2^-}{k_2^+} + \left(\frac{c_1^+ + c_1^-}{c_1^+} + \frac{c_4^+ + c_4^-}{c_4^+} \frac{k_4^-}{k_4^+} \right) \frac{k_2^+ k_3^+ + k_1^- k_3^+ + k_1^- k_2^-}{k_1^+ k_2^+}},$$

where $\lim\limits_{t\to\infty} T_i G(t)$ means the value of $T_i G(t)$ at steady–state of system (1).

Corresponding leading order approximation has the form:

$$\Xi_0 = \frac{m_2^+ m_3^+ m_4^+}{m_2^+(m_3^+ + m_4^+) + m_4^+(m_2^- + m_3^+) + \frac{m_4^+ + m_4^-}{m_1^+}(m_1^- m_2^- + m_1^- m_3^+ + m_2^+ m_3^+)}, \quad (10)$$

which can be found by approximating $T_i G(t)$ using (6). Note that the approximation is written in terms of the parameters of the system (5) and none of the original parameters of the system (1) enter the equation (10).

3.1.3 Experiment and model implementation

The described experiments were performed. We injected stage V–VI *Xenopus* oocytes with mRNA encoding one of EAAT1—3 (the experiments were repeated for each transporter subtype). We waited for 2–5 days to allow oocytes to express the transporters and excised outside-out patches. In the outside-out patches experiments, the solution inside the pipette represents cell's cytosol. The solution we made contained $110mM\ KCl$, $3mM\ MgCl_2$, $5mMNa - HEPES$, and $10mM\ EGTA$. The bath solutions, which correspond to extracellular space, contained $110mM\ NaCl$, $3mM\ MgCl_2$, $5mM\ Na - HEPES$, and either $10mM$ or no glutamate. We adjusted all solutions to Ph 7.5 with Tris—base (a common buffer solution used in biochemistry). Both bath mixtures flowed through a theta tube that was mounted on and controlled by a piezo electric switch. We voltage clamped the cells at $-60mV$, which is close to a natural resting potential of neurons and astrocytes. For more details of the voltage clamp technique see [9]. As described, each experiment started with the patch exposed to a bath solution with no glutamate and was allowed to stabilize and reach current steady–state. After that we did a short rapid $50ms$ step into a solution with glutamate before switching back to the no—glutamate mixture. Then we waited for a varied controlled amount of time (5, 10, 15, 20, 30, 40, 50, 60, 80, 100, 150, 200, 250, and 300 ms) and applied a second $30ms$ step into the glutamate containing solution. Note that the only differences between the experimental trials was a delay after the first pulse. See Figure 5 for the representative data.

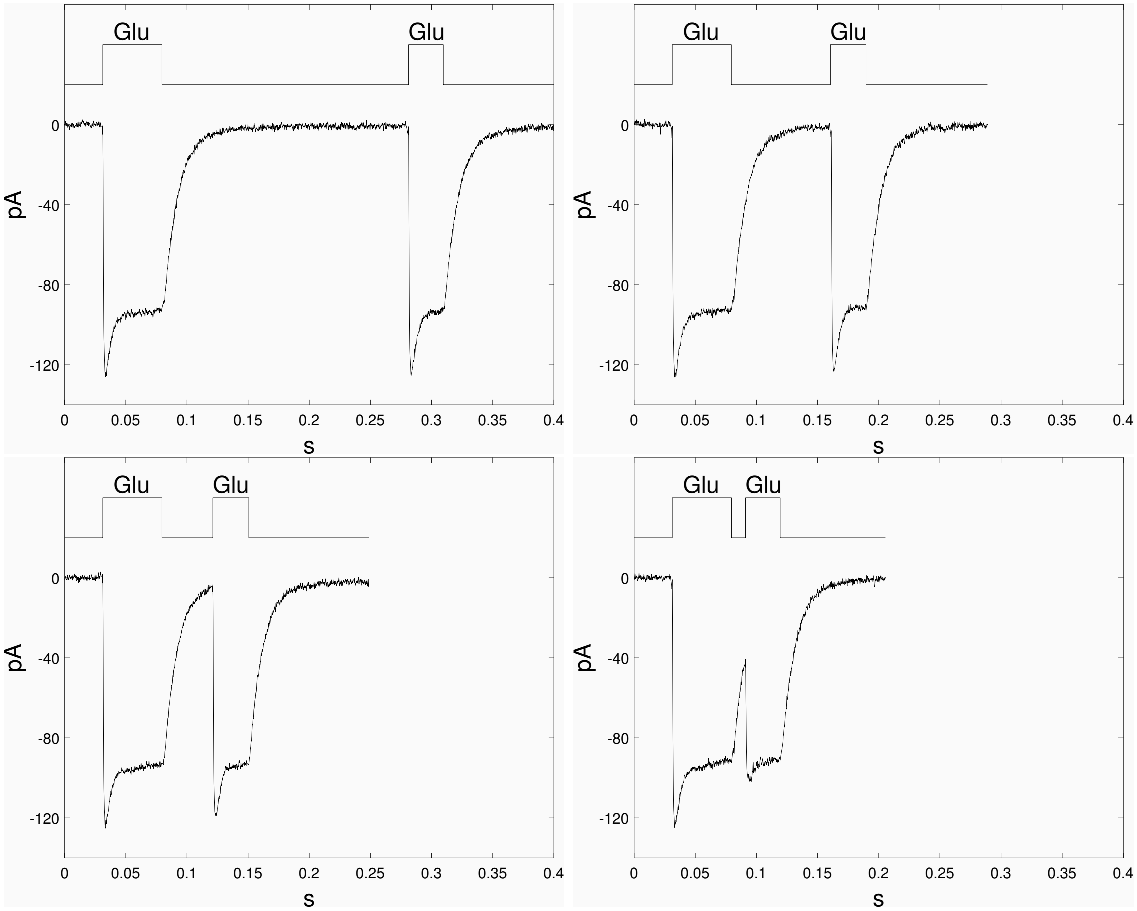

Figure 5: The currents recorded during the voltage clamp experiments in patches expressing EAAT1. The time courses of glutamate application are depicted as step functions above the actual data. Each application of the glutamate caused a spike of a negative (inward) current that relaxed to a new steady–state. After the termination of the glutamate pulse, the system tended to its original steady–state. The delays between the pulses depicted here are 200, 80, 40, and 10 ms.

The data for each transporter were fitted to model (5), (8), and (9) using the delayed rejection adaptive Metropolis Markov chain Monte Carlo (DRAM MCMC) method [19]. The fitting was done in MATLAB R2018a using MCMC toolbox [20] (see the code in Appendix B.1). The fitted model solutions practically coincide with the experimental data for all cases, see Figure 6. The algorithm yielded empirical confidence regions for the parameters; the resulting MCMC chains were used for turnover rate estimates and inferences using the formula (10); see Figures 7 — 9 for confidence regions.

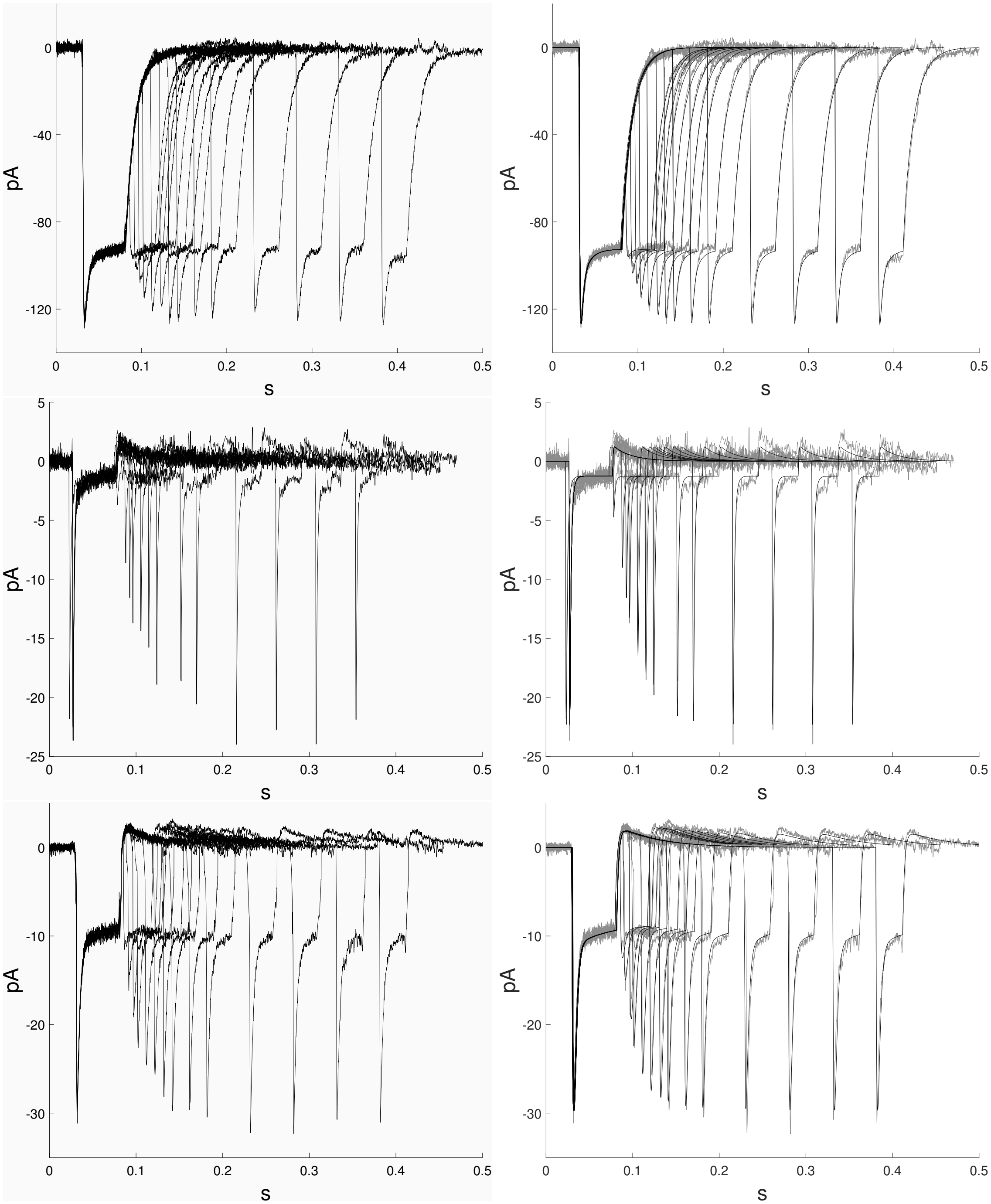

Figure 6: The currents recorded during the voltage clamp experiments in excised outside-out patches were fitted to the model. First column depicts all the data for the corresponding transporter combined (top to bottom: EAAT1, EAAT2, EAAT3). All recorded pairs of pulses like the ones in Figure 5 were overlapped and the first pulses being almost identical. With the first pulse being fixed, it is easy to see the change in the dynamics of the second pulses, e.g., note how the peaks of the second pulses deteriorate but recover as the delay between the pulses increases. The second column shows the same data (now in gray) fitted using the model (black curves).

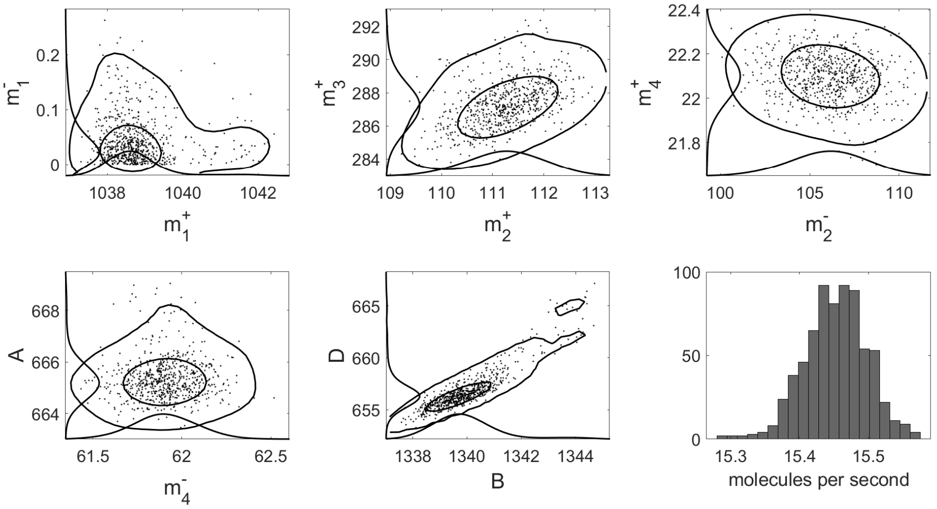

Figure 7: 95% and 99% confidence regions (inner and outer contours, respectively) for the model of EAAT1 yielded by the MCMC method. The curves along the axes represent projection of the data on the corresponding axis. The last picture shows the distribution of EAAT1 turnover rate with the median value of $15.45s^{-1}$, 99% confidence interval: $[15.31, 15.57]$.

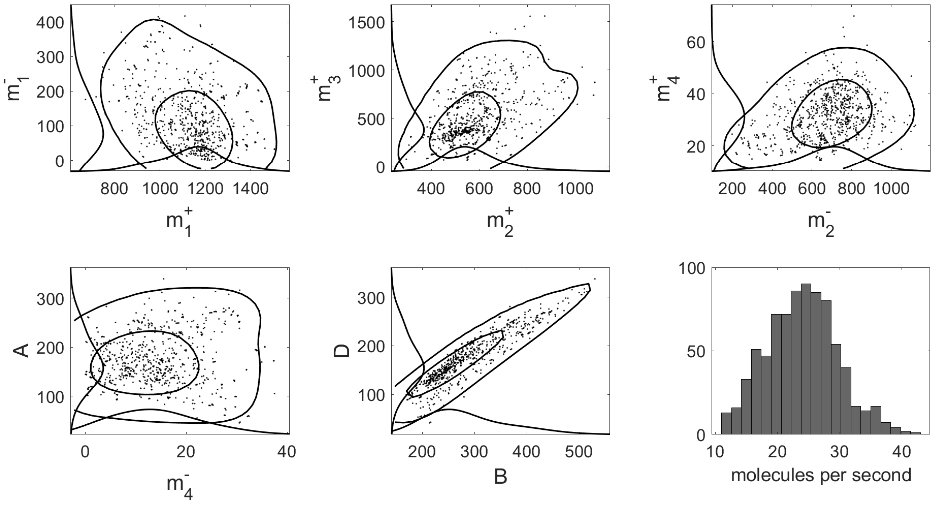

Figure 8: Similar results are shown for EAAT2. The median turnover rate value is $23.96s^{-1}$, 99% confidence interval: $[11.94, 39.55]$.

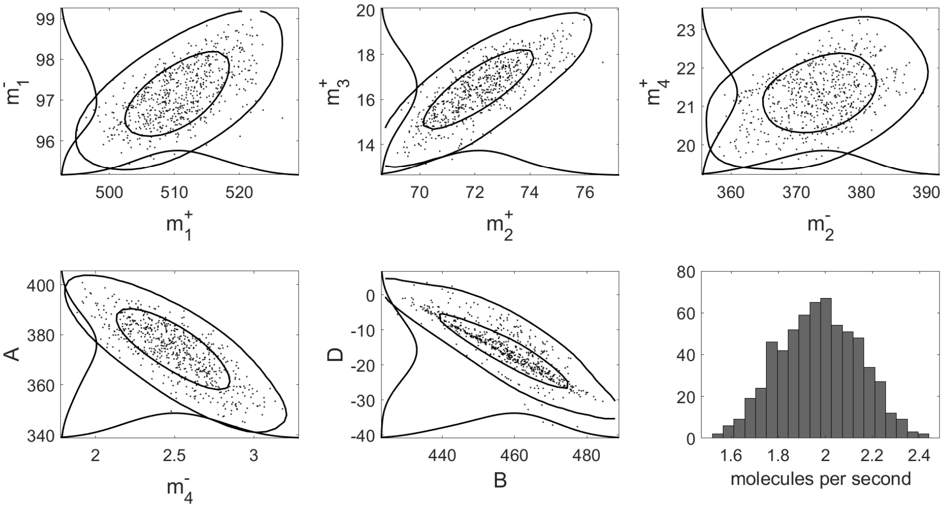

Figure 9: Similar results are shown for EAAT3. The estimated EAAT3 turnover rate of $1.98s^{-1}$, 99% confidence interval: $[1.57, 2.36]$, was unexpectedly small compared to hundreds of molecules per second reported in other literature.

The estimated value for EAAT1 turnover rate found here ($15.45s^{-1}$, 99% confidence interval $[15.31, 15.57]$) is in agreement with previous studies: 16 molecules per second in EAAT1 [9]. The estimate for EAAT2 turnover rate ($23.96s^{-1}$, 99% confidence interval $[11.94, 39.55]$) is close to some previous studies, e.g., 14.6 molecules per second in [8], while it is lower than $\sim$ 100 per second reported in [11]. The turnover rate of EAAT3 was reported to be much higher: about 100 molecules per second [10], compared to our estimate ($1.98s^{-1}$, 99% confidence interval $[1.57, 2.36]$).

3.2 Regulation of glutamate concentration in synaptic transmission

3.2.1 Prolongation of synaptic transmission

The EAAT turnover rates estimates obtained in Section 3.1 are close to the ones in the existing literature. They are significantly slower than the estimated millisecond presence of glutamate in the synaptic cleft during the transmission [21, 22]. It follows that they should not play a significant role during the signaling itself but rather help to clear the synaptic cleft on a slower time scale. However, experiments in hippocampal slices show that when the transporters are blocked with DL-threo-benzyloxyaspartic acid (TBOA), a competitive EAAT antagonist, this seems to prolong glutamate receptor activation during synaptic transmission [23, 24, 25, 26, 27]. These fast changes would require much higher EAATs expression in the synaptic cleft than reported [6]. It is worth noting that all these experiments were performed in a whole cell or multiple cell recording settings, i.e. the data was acquired from accumulated effect of activity of many synapses. It is possible that the apparent prolongation is due to some other effects. For example, the application of blockers raises ambient glutamate level which activates extrasynaptic receptors [28]. These receptors could play a significant role in apparent synaptic prolongation. In order to understand the relative contribution of transport and diffusion to dynamics of glutamate we create a spatial model of a single synapse and simulate a synaptic transmission.

3.2.2 Model derivation

We construct a 3—dimensional diffusion model of the synapse. Let us represent a synaptic cleft as a circular cylinder, with bases being pre— and postsynaptic cell surfaces, and let us describe the glutamate dynamics using a diffusion model. During a synaptic transmission upon arrival of an action potential to the axon terminal of the presynaptic cell, the intracellular vesicles containing glutamate merge with the membrane of the cell and release glutamate into the synaptic cleft. The release is mediated by a specialized region on a presynaptic cell, called the active zone (AZ). The neurotransmitter diffuses across the synaptic cleft and reaches receptors on the postsynaptic cell, located in the region called the postsynaptic density (PSD). We model AZ and PSD using circular shapes with their centers coinciding with the corresponding cylinder base centers. About 50% of the synapses in rat hippocampus region CA1 are enveloped by astrocytes and they are surrounded only partially [5]. The synapse's ensheathment by the astrocyte limits the escape of a glutamate from the synaptic cleft via free diffusion. In our model the astrocyte partially covers the sides of the cylinder. Some of EAATs are expressed in a synaptic cleft, while others are found on astrocytes: EAAT3 are exclusively expressed on neurons' in dendrites and somas and EAAT1 are found on astrocytes. EAAT2 are expressed on both astrocytes and synapses of presynaptic cells, although EAAT2's expression in synapses is about 10 times lower [6, 7]. We assume that the distribution of EAATs is uniform where they are present. In terms of our model, the presence of EAATs will determine the boundary conditions of the region. We note once again that this model describes a single synapse, so we assume that the vesicular glutamate is the only source of the glutamate (this is not always the case in reality, e.g., when glutamate diffuses away from a synapse it can reach neighboring synapses and activate them indirectly; this phenomenon called a spillover). See Figure 10 for a visual representation of the described model.

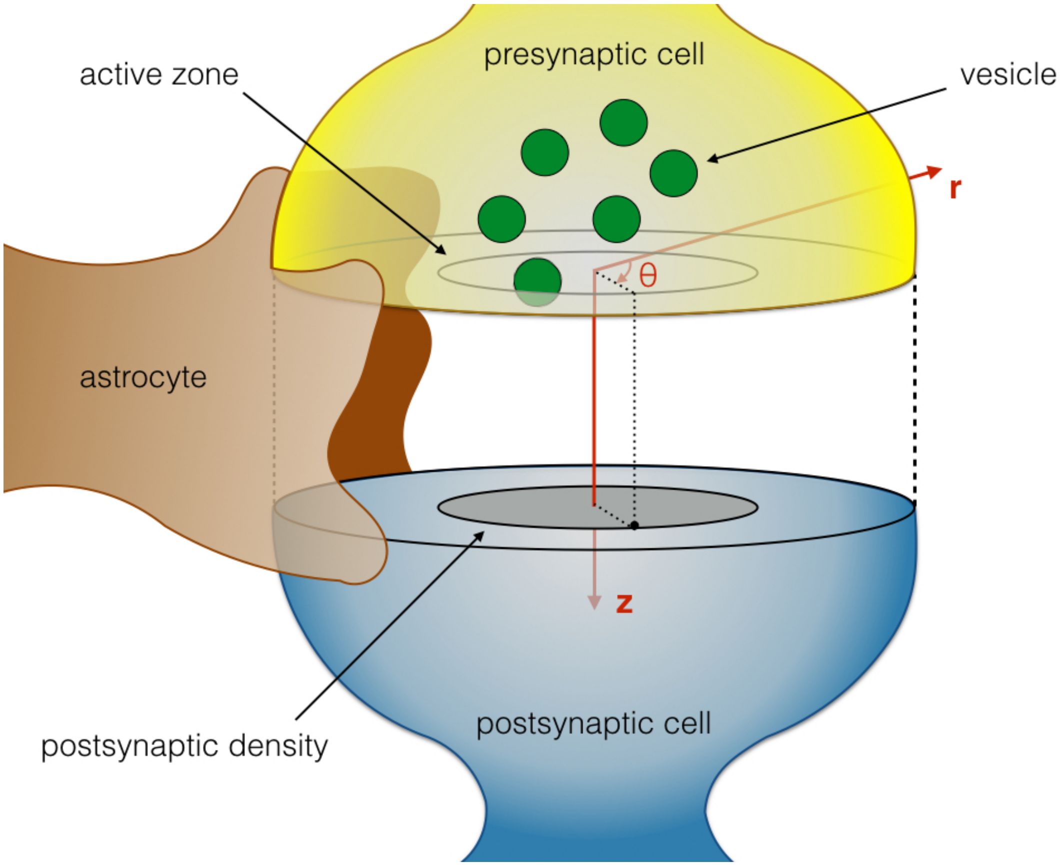

Figure 10: A cartoon representation of the synapse. The presynapic cell contains vesicles (green) with glutamate. Upon the arrival of the action potential they merge with the membrane of the presynaptic cell (yellow) at the active zone (inner circle). This causes the glutamate to diffuse into the synaptic cleft and eventually reach the receptors of the postsynaptic cell (blue). The receptors are located in the postsynaptic density (inner circle) of the postsynaptic cell. Astrocyte (brown) envelops the cleft to some degree that varies from synapse to synapse. EAAT1—2 transporters are located on the astrocyte facing the synapse. EAAT2 also found in the cleft on the presynaptic cell outside of the active zone with expression ∼ 10 times lower compared to the one in astrocytes. EAAT3 are expressed on the postsynaptic cell outside of the postsynaptic density.

Let us write a diffusion equation in cylindrical coordinates:

$$\frac{\partial u}{\partial t} = D\left(\frac{1}{r}\frac{\partial}{\partial r}\left[r\frac{\partial u}{\partial r}\right] + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial z^2}\right), \quad t \geq 0, \quad 0 < r \leq R, \quad 0 \leq \theta < 2\pi, \quad 0 \leq z \leq L, \tag{11}$$

where $u(t, r, \theta, z)$ is a concentration of glutamate at a time t, at a radial distance r, angular coordinate θ and height z; $D = 0.4\mu m^2 ms^{-1}$ is the diffusion coefficient of glutamate in a synaptic cleft, which is about half of that in a free medium [29, 30, 31]; $R = 150nm$ is the radius of a synaptic cleft [5, 30]; $L = 20nm$ is the width of a synaptic cleft [30]. Due to the fact that an astrocyte does not cover the synapse symmetrically, we cannot reduce the dimensions of the system. That makes the equation (11) being undefined when $r = 0$ as we cannot simply use the L'Hôpital's rule here. At $r = 0$, the equation can be rewritten in Cartesian coordinates:

$$\frac{\partial u_C}{\partial t} = D\left(\frac{\partial^2 u_C}{\partial x^2} + \frac{\partial^2 u_C}{\partial y^2} + \frac{\partial^2 u_C}{\partial z^2}\right), \quad t \geq 0, \quad r = 0, \quad 0 \leq \theta < 2\pi, \quad 0 \leq z \leq L, \tag{12}$$

where u_C is just the same function u in the Cartesian coordinates.

We performed the EAATs' dynamics modeling in Section 3.1. And although it is straightforward to implement the corresponding chemical—kinetics systems into the diffusion equation (Appendix A.2), we cannot do this here directly. We can use neither of the models depicted in the Figures 2, 3 because, as we mentioned previously, the estimates of their parameters are unreliable. We also cannot use the model we derived (Figure 4) because it assumes the constant saturating concentration of the glutamate whenever it is present. It is widely accepted that EAATs follow a Michaelis—Menten kinetics with known parameters [6, 32]. Thus, for each transporter the rate of the uptake of the glutamate is described by

$$V\frac{[Glu]}{[Glu] + K_m},$$

where V and K_m are a maximal turnover rate and a known Michaelis—Menten constant for a particular type (1—3) of the transporter, respectively; $[Glu]$ is a concentration of the glutamate in the thin layer near a cell surface containing transporters. Since we reliably estimated the maximal turnover rates of the transporters in Section 3.1, we can use the Michaelis—Menten approximation of the transporter's dynamic in the boundary conditions for our diffusion model. We define V_1, V_2, V_3 as turnover rates of EAAT1—3 found in Section 3.1, respectively. We also define N_1, N_2, N_2' N_3 as surface densities of EAAT1 on an astrocyte, EAAT2 on an astrocyte, EAAT2 on a presynaptic neuron, EAAT3 on a postsynaptic neuron, respectively. And $K_m^{(1)}$, $K_m^{(2)}$, $K_m^{(3)}$ are EAAT1—3 Michaelis—Menten constants, respectively. If there were no tonic glutamate present in the extracellular space, then the boundary conditions would be:

$$
\begin{aligned}
D\left.\frac{\partial u}{\partial z}\right|_{z=0} &= \begin{cases} 0, & 0 \le r \le r_a, \quad \forall\theta, \\ N_2'V_2 \dfrac{u(t,r,\theta,0)}{K_m^{(2)} + u(t,r,\theta,0)}, & r_a < r \le R, \quad \forall\theta, \end{cases} \\
-D\left.\frac{\partial u}{\partial z}\right|_{z=L} &= \begin{cases} 0, & 0 \le r \le r_d, \quad \forall\theta, \\ N_3V_3 \dfrac{u(t,r,\theta,L)}{K_m^{(3)} + u(t,r,\theta,L)}, & r_d < r \le R, \quad \forall\theta. \end{cases}
\end{aligned}
$$

We note that the parameter that corresponds to the width of the layer where glutamate interacts with transporters (ω in Appendix A.2) disappears as volume densities of transporters become surface densities. The ambient glutamate level at the steady–state is $u_\infty = 25nM$ [28, 33, 34, 35, 36], and we want fluxes to be zero when $u = u_\infty$. Thus, the boundary conditions must be slightly modified to force this requirement:

$$
\begin{aligned}
D\left.\frac{\partial u}{\partial z}\right|_{z=0} &= \begin{cases} 0, & 0 \le r \le r_a, \quad \forall\theta, \\ N_2'V_2 \left(\dfrac{u(t,r,\theta,0)}{K_m^{(2)} + u(t,r,\theta,0)} \right. & \\ \qquad\qquad \left. - \dfrac{u_\infty}{K_m^{(2)} + u_\infty} \right), & r_a < r \le R, \quad \forall\theta, \end{cases} \\
-D\left.\frac{\partial u}{\partial z}\right|_{z=L} &= \begin{cases} 0, & 0 \le r \le r_d, \quad \forall\theta, \\ N_3V_3 \left(\dfrac{u(t,r,\theta,L)}{K_m^{(3)} + u(t,r,\theta,L)} \right. & \\ \qquad\qquad \left. - \dfrac{u_\infty}{K_m^{(3)} + u_\infty} \right), & r_d < r \le R, \quad \forall\theta. \end{cases}
\end{aligned}
\tag{13}
$$

The last boundary condition is:

$$
-D\left.\frac{\partial u}{\partial r}\right|_{r=R} = \sum_{n=1}^{2} N_nV_n \left(\frac{u(t,R,\theta,z)}{K_m^{(n)} + u(t,R,\theta,z)} - \frac{u_\infty}{K_m^{(n)} + u_\infty} \right), \qquad 0 \le z \le L, \quad 0 \le \theta < 2\pi\rho,
\tag{14}
$$

which is formulated for the proportion of synapse covered by astrocyte, defined by the parameter $0 \le \rho \le 1$, and

$$
u(t,R,\theta,z) = u_\infty, \quad 2\pi\rho \le \theta < 2\pi.
\tag{15}
$$

Here $r_a = 100nm$ is the radius of an active zone [37]; $r_d = 100nm$ is the radius of a postsynaptic density [5, 30]; $N_1 = 2300\mu m^{-2}$, $N_2' = 750\mu m^{-2}$, $N_2 = 7500\mu m^{-2}$, and $N_3 = 90\mu m^{-2}$ are the densities of EAAT1, EAAT2 on axon terminals, EAAT2 on astrocytes, EAAT3, respectively [6, 7]; $V_1 = 1.545 \times 10^{-2} ms^{-1}$, $V_2 = 2.396 \times 10^{-2} ms^{-1}$, and $V_1 = 0.198 \times 10^{-2} ms^{-1}$ are the EAAT1—3 turnover rates, respectively. Glutamate uptake by the transporters is described by Michaelis-Menten kinetics with $K_m^{(1)} = 20\mu M$, $K_m^{(2)} = 18\mu M$, and $K_m^{(3)} = 28\mu M$ constants for EAAT1—3, respectively [6, 32].

We model the synaptic transmission starting with the moment when glutamate is released from the presynaptic cell. Not every action potential in a presynaptic cell causes the release of even a single vesicle [5, 38], which contain about 3000 molecules [30, 39]. To conservatively estimate the clearance of a synaptic cleft from the glutamate, we assume that exactly one vesicle is released for each stimulation from the center of the active zone. For the sake of simplicity we assume that the vesicle is released from the center of AZ, i.e., the initial conditions could be modeled as

$$u(0, r, \theta, z) = 3000 \cdot \delta(r\cos\theta) \cdot \delta(r\sin\theta) \cdot \delta(z) + w(r, \theta, z), \tag{16}$$

where $\delta(\cdot)$ is the Dirac delta function and w is a state of the system before the pulse. We start with $w(r, \theta, z) \equiv u_\infty$ and if we want to simulate several stimulation in a row, then $w(r, \theta, z)$ is going to be equal to the end state of the model just before the consecutive new release of the glutamate.

We are interested in the dynamics of glutamate concentration on the surface of the PSD of the postsynaptic cell, i.e, in the average glutamate concentrations on the surface of the postsynaptic density "PSD" according to the diffusion model.

3.2.3 Model discretization

We will numerically solve the diffusion equation (11)—(12) with nonlinear boundary conditions (13)—(15) and initial conditions (16) by applying a finite difference discretization method (Appendix A.3) to the spatial variables r, θ, z (or x, y, z for (12)), which yields a system of ordinary differential equations. This allows us to use a built—in MATLAB solver for ODEs with an adaptive time step. The latter is desirable since the experimental data clearly show relatively fast and slow dynamics.

Let us introduce the following grids:

$$\begin{aligned} r_i &= i \cdot \Delta r, & \Delta r &= \frac{R}{I}, & i &= 0, \ldots, I, \\ \theta_j &= j \cdot \Delta\theta, & \Delta\theta &= \frac{2\pi}{J+1}, & j &= 0, \ldots, J, \\ z &= k \cdot \Delta z, & \Delta z &= \frac{L}{K}, & j &= 0, \ldots, K. \end{aligned}$$

and $u_{i,j,k}(t) = u(t, r_i, \theta_j, z_k)$. Moreover, we make the grid for θ in such a way that there are indices j_0, j_1, j_2, and j_3 that $\theta_{j_0} = 0$, $\theta_{j_1} = \pi/2$, $\theta_{j_2} = \pi$, and $\theta_{j_3} = 3\pi/2$ (it is obvious that $j_0 = 0$ always). Then we can use (11) to write the corresponding equation for $u_{i,j,k}$ in case $i > 0$:

$$\begin{aligned}\frac{1}{D}\frac{du_{i,j,k}}{dt} &= \frac{1}{\Delta r \cdot i}\frac{u_{i+1,j,k} - u_{i-1,j,k}}{2\Delta r} + \frac{u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}}{\Delta r^2} \\ &+ \frac{1}{(\Delta r \cdot i)^2}\frac{u_{i,j+1,k} - 2u_{i,j,k} + u_{i,j-1,k}}{\Delta \theta} + \frac{u_{i,j,k+1} - 2u_{i,j,k} + u_{i,j,k-1}}{\Delta z^2}.\end{aligned} \tag{17}$$

Let us note that the values of $u_{i,j,k}$ are not technically defined outside of the region boundaries, e.g., $u_{I+1,j,k}$ (which is used in equation for $u_{I,j,k}$), but they can be found from the boundary conditions. We also have a special case when $r = 0$. First of all, $u_{0,j,k}$ are the same for all j's and any fixed k. Second, in this case we must use (12) rather than (11). We write (using the same notations):

$$\begin{aligned}\frac{1}{D}\frac{du_{0,j,k}}{dt} = \frac{u_{1,j_0,k} - 2u_{0,j,k} + u_{1,j_2,k}}{\Delta r^2} &+ \frac{u_{1,j_1,k} - 2u_{0,j,k} + u_{1,j_3,k}}{\Delta r^2} \\ &+ \frac{u_{0,j,k+1} - 2u_{0,j,k} + u_{0,j,k-1}}{\Delta z^2}.\end{aligned} \tag{18}$$

Since θ is an angular coordinate, we have

$$u_{i,J+1,k} = u_{i,0,k}, \qquad u_{i,-1,k} = u_{i,J,k}, \qquad \forall i, k. \tag{19}$$

From (13) we can find $u_{i,j,-1}$ and $u_{i,j,K+1}$. Let us use the first equation of (13):

$$D\frac{u_{i,j,1} - u_{i,j,-1}}{2\Delta z} = \begin{cases} 0, & 0 \le i \le \left[\frac{r_a}{\Delta r}\right], \quad \forall j, \\ N_2' V_2 \left(\frac{u_{i,j,0}}{K_m^{(2)} + u_{i,j,0}} - \frac{u_\infty}{K_m^{(2)} + u_\infty}\right), & \left[\frac{r_a}{\Delta r}\right] < i \le I, \quad \forall j. \end{cases}$$

where $[\cdot]$ is the operation of rounding to the nearest integer. So,

$$u_{i,j,-1} = \begin{cases} u_{i,j,1}, & 0 \le i \le \left[\frac{r_a}{\Delta r}\right], \quad \forall j, \\ u_{i,j,1} - \frac{2\Delta z N_2' V_2}{D}\left(\frac{u_{i,j,0}}{K_m^{(2)} + u_{i,j,0}} - \frac{u_\infty}{K_m^{(2)} + u_\infty}\right), & \left[\frac{r_a}{\Delta r}\right] < i \le I, \quad \forall j. \end{cases} \tag{20}$$

Analogously

$$u_{i,j,K+1} = \begin{cases} u_{i,j,K-1}, & 0 \le i \le \left[\dfrac{r_d}{\Delta r}\right], \quad \forall j, \\ u_{i,j,K-1} - \dfrac{2\Delta z N_3 V_3}{D}\left(\dfrac{u_{i,j,K}}{K_m^{(3)} + u_{i,j,K}} \right. & \\ \qquad\qquad \left. - \dfrac{u_\infty}{K_m^{(3)} + u_\infty}\right), & \left[\dfrac{r_d}{\Delta r}\right] < i \le I, \quad \forall j, \end{cases} \tag{21}$$

and

$$u_{I+1,j,k} = \begin{cases} u_\infty, & 0 \le j \le [2\pi\rho], \quad \forall k, \\ u_{I-1,j,k} - \dfrac{2\Delta r}{D}\sum_{n=1}^{2}\left(\dfrac{u_{I,j,k}}{K_m^{(n)} + u_{I,j,k}} \right. & \\ \qquad\qquad \left. - \dfrac{u_\infty}{K_m^{(n)} + u_\infty}\right), & [2\pi\rho] < j \le J, \quad \forall k. \end{cases} \tag{22}$$

Since we are using MATLAB, it is beneficial to write the equations in matrix form. To do that, we define a 3—dimensional array U of size $(I+1) \times (J+1) \times (K+1)$ in the following manner. Let $U_{:,:,k}$, $U_{:,j,:}$, and $U_{i,:,:}$ be the 2—dimensional slices of U at level k of the third dimension and parallel to the first and second dimensions, at level j of the second dimension and parallel to the first and third dimensions, and at level i of the first dimension and parallel to the second and third dimensions, respectively. Then these slices may be written as follows:

$$U_{:,:,k} = \begin{bmatrix} u_{0,0,k} & \cdots & u_{0,J,k} \\ \vdots & \ddots & \vdots \\ u_{I,0,k} & \cdots & u_{I,J,k} \end{bmatrix}, \quad U_{:,j,:} = \begin{bmatrix} u_{0,j,0} & \cdots & u_{0,j,K} \\ \vdots & \ddots & \vdots \\ u_{I,j,0} & \cdots & u_{I,j,K} \end{bmatrix},$$

$$U_{i,:,:} = \begin{bmatrix} u_{i,0,0} & \cdots & u_{i,0,K} \\ \vdots & \ddots & \vdots \\ u_{i,J,0} & \cdots & u_{i,J,K} \end{bmatrix}.$$

We will also start the indices with 0 for convenience. One could imagine U as a stack of its corresponding slices. For example, a stack of $U_{:,:,k}$ slices ($k = \overline{0, K}$) standing one after another in the third dimension, with matrix $U_{:,:,0}$ being the frontmost and matrix $U_{:,:,K}$ being the backmost. We consider U to be a simple extension of the regular matrices and define a matrix multiplication. Note that we do not want to introduce the whole concept of tensors here as it is not really needed. Next, we define array multiplication. Suppose F

is a $P \times (I+1) \times (K+1)$ array, then we say that U can be multiplied by F from the left and their product is a $P \times (J+1) \times (K+1)$ array that is defined as

$$(F * U)_{:,:,k} = F_{:,:,k} \cdot U_{:,:,k}, \qquad k = \overline{0, K}.$$

Suppose G is a $(J+1) \times Q \times (K+1)$ array, then we say that U can be multiplied by G from the right and their product is an $(I+1) \times Q \times (K+1)$ array that is defined as

$$(U * G)_{:,:,k} = U_{:,:,k} \cdot G_{:,:,k}, \qquad k = \overline{0, K}.$$

Suppose H is an $S \times (J+1) \times (I+1)$ array, then we say that U can be multiplied by H from the front and their product is an $S \times (J+1) \times (K+1)$ array that is defined as

$$(H \star U)_{:,j,:} = H_{:,j,:} \cdot U_{:,j,:}, \qquad j = \overline{0, J}.$$

Suppose P is a $(K+1) \times (J+1) \times W$ array, then we say that U can be multiplied by H from the back and their product is an $(I+1) \times (J+1) \times W$ array that is define as

$$(U \star P)_{:,j,:} = U_{:,j,:} \cdot P_{:,j,:}, \qquad j = \overline{0, J}.$$

Note, that "$\cdot$" stands for a standard matrix multiplication. See Figure 11 for a visual representation of 3—dimensional array. Now, we are ready to write the discretized form of (11):

$$\frac{1}{D}\frac{dU}{dt} = \frac{1}{\Delta r^2} A * U + \left(\frac{1}{\Delta r^2 \cdot \Delta \theta^2} U * B \right) \circ B' + \frac{1}{\Delta z^2} U \star C + E, \tag{23}$$

where "$\circ$" is the Hadamard product. We remind that "$*$" is used for multiplication from the left and from the right, and "$\star$" is used for multiplication from the front and from the back. And matrices A, B, B', C are the following. Array A is $(I+1) \times (I+1) \times (K+1)$ and its slices $A_{:,:,k}$ are the same across and are defined as

$$A_{:,:,k} = \begin{bmatrix} -4 & \beta_0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \gamma_1 & -2 & \beta_1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \gamma_2 & -2 & \beta_2 & \cdots & 0 & 0 & 0 \\ \vdots & & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \gamma_{I-1} & -2 & \beta_{I-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & \gamma_I & -2 \end{bmatrix},$$

where

$$\begin{aligned} \beta_i &= \begin{cases} 0, & i = 0, \\ 1 + \dfrac{1}{2i}, & i = \overline{1, I}, \end{cases} \\ \gamma_i &= 1 - \frac{1}{2i}, \qquad i = \overline{1, I}. \end{aligned}$$

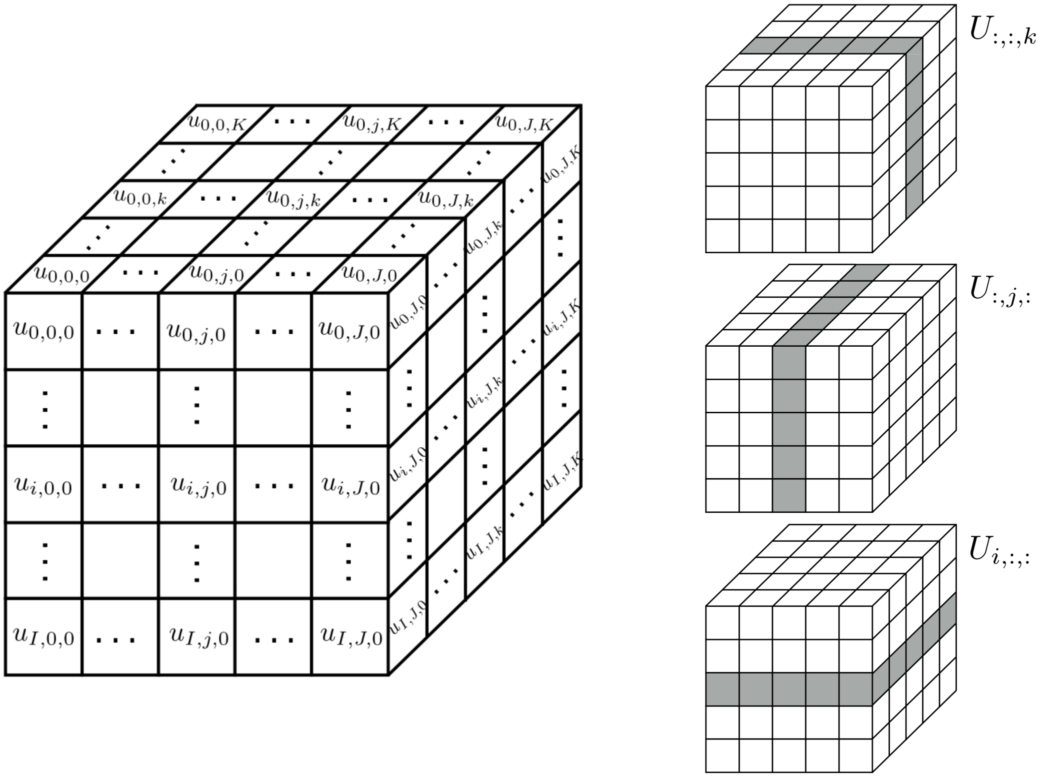
$u_{0,0,K}$ ⋯ $u_{0,j,K}$ ⋯ $u_{0,J,K}$
$u_{0,0,k}$ ⋯ $u_{0,j,k}$ ⋯ $u_{0,J,k}$
$u_{0,0,0}$ ⋯ $u_{0,j,0}$ ⋯ $u_{0,J,0}$
$u_{0,0,0}$ ⋯ $u_{0,j,0}$ ⋯ $u_{0,J,0}$
$u_{i,0,0}$ ⋯ $u_{i,j,0}$ ⋯ $u_{i,J,0}$
$u_{I,0,0}$ ⋯ $u_{I,j,0}$ ⋯ $u_{I,J,0}$
$u_{0,J,0}$
$u_{i,J,0}$
$u_{I,J,0}$
$u_{0,J,k}$
$u_{i,J,k}$
$u_{I,J,k}$
$u_{0,J,K}$
$u_{i,J,K}$
$u_{I,J,K}$
$U_{:,:,k}$
$U_{:,j,:}$
$U_{i,:,:}$

Figure 11: A 3—dimensional array U and its slices. Each slice is a usual 2—dimensional matrix. Multiplication of 3—dimensional arrays is a multiplication of corresponding slices using rules for a regular 2—dimensional matrix multiplication and stacking the resulting matrices together ($U_{:,:,k}$ for the multiplication from the left and from the right; $U_{:,j,:}$ for the multiplication from the front and from the back; we do not consider multiplications from the top or bottom here.

Array B is $(J+1)\times(J+1)\times(K+1)$ and its slices $B_{:,:,k}$ are the same across and are defined as

$$B_{:,:,k} = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 \\ 0 & 1 & -2 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -2 & 1 \\ 0 & 0 & 0 & \cdots & 1 & -2 \end{bmatrix}.$$

Array B' is $(I+1)\times(J+1)\times(K+1)$ and its slices $B'_{:,:,k}$ are the same across and are defined as

$$B'_{:,:,k} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \frac{1}{1^2} & \frac{1}{1^2} & \cdots & \frac{1}{1^2} \\ \frac{1}{2^2} & \frac{1}{2^2} & \cdots & \frac{1}{2^2} \\ \vdots & \vdots & & \vdots \\ \frac{1}{I^2} & \frac{1}{I^2} & \cdots & \frac{1}{I^2} \end{bmatrix}.$$

Array C is $(K+1)\times(J+1)\times(K+1)$ and its slices $C_{:,j,:}$ are the same across and are defined as

$$C_{:,j,:} = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 \\ 0 & 1 & -2 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -2 & 1 \\ 0 & 0 & 0 & \cdots & 1 & -2 \end{bmatrix}.$$

Array E is $(I+1)\times(J+1)\times(K+1)$ and its purpose is to add the missing terms for the "boundary" elements. It is convenient to represent array E as a sum of the arrays of the same size:

$$E = E^{(1)} + E^{(2)} + E^{(3)} + E^{(4)},$$

where $E^{(1)}$ corresponds to the boundary conditions for z variables and is an array with all entries being zero except the for the frontmost and the backmost slices:

$$E^{(1)}_{:,:,0} = \frac{1}{\Delta z^2} \begin{bmatrix} u_{0,0,-1} & \cdots & u_{0,J,-1} \\ \vdots & \ddots & \cdots \\ u_{I,0,-1} & \cdots & u_{I,J,-1} \end{bmatrix},$$

$$E^{(1)}_{:,:,K} = \frac{1}{\Delta z^2} \begin{bmatrix} u_{0,0,K+1} & \cdots & u_{0,J,K+1} \\ \vdots & \ddots & \cdots \\ u_{I,0,K+1} & \cdots & u_{I,J,K+1} \end{bmatrix},$$

with all entries defined by (20) and (21); $E^{(2)}$ corresponds to the boundary conditions for θ and is an array with all entries being zero except for the leftmost and rightmost slices:

$$E^{(2)}_{:,0,:} = \frac{1}{\Delta r^2 \Delta\theta^2} \begin{bmatrix} u_{0,-1,0} & \cdots & u_{0,-1,K} \\ \vdots & \ddots & \vdots \\ u_{I,-1,0} & \cdots & u_{I,-1,K} \end{bmatrix} \circ B'_{:,:,0},$$

$$E^{(2)}_{:,J,:} = \frac{1}{\Delta r^2 \Delta\theta^2} \begin{bmatrix} u_{0,J+1,0} & \cdots & u_{0,J+1,K} \\ \vdots & \ddots & \vdots \\ u_{I,J+1,0} & \cdots & u_{I,J+1,K} \end{bmatrix} \circ B'_{:,:,0},$$

with all entries defined by (19) and ”$\circ$” being a Hadamard product; $E^{(3)}$ corresponds to the boundary conditions for r and is an array with all entries being zero except for the bottommost slice:

$$E^{(3)}_{R,:,:} = \frac{1}{\Delta r^2} \left(\frac{1}{2I} + 1 \right) \begin{bmatrix} u_{I+1,0,0} & \cdots & u_{I+1,0,K} \\ \vdots & \ddots & \vdots \\ u_{I+1,J,0} & \cdots & u_{I+1,J,K} \end{bmatrix},$$

with all entries defined by (22); $E^{(4)}$ corresponds to a special case when $r = 0$, described by the equation (18), and is an array with all entries being zero except for the topmost slice:

$$E^{(4)}_{0,:,:} = \frac{1}{\Delta r^2} \begin{bmatrix} \sum\limits_{s=0}^{3} u_{1,j_s,0} & \sum\limits_{s=0}^{3} u_{1,j_s,1} & \cdots & \sum\limits_{s=0}^{3} u_{1,j_s,K} \\ \vdots & \vdots & \vdots & \vdots \\ \sum\limits_{s=0}^{3} u_{1,j_s,0} & \sum\limits_{s=0}^{3} u_{1,j_s,1} & \cdots & \sum\limits_{s=0}^{3} u_{1,j_s,K} \end{bmatrix}.$$

The initial conditions (16) become

$$U = W + \overline{U}, \tag{24}$$

where W is the end state of the previous solution or a 3—dimensional array with all entries being the same constant u_∞ in the case of the first pulse; $\overline{U}$ is an array with all entries being zero except for one: $\overline{U}_{0,0,0} = 3000 \cdot 4/(\pi \Delta r^2 \Delta z)$ molecules per nm^3.

3.2.4 Simulation

We convert all units into nm for length; ms for time; # for amount of molecules, that we define as number of molecules multiplied by a scale factor of 10^6 (to avoid numbers going below machine precision). Here is the list of all numerical values used in the simulation (see code in Appendix B.2):

$$
\begin{array}{rcl c rcl}
L & = & 20\ nm, & & R & = & 150\ nm, \\
r_a & = & 100\ nm, & & r_d & = & 100\ nm, \\
D & = & 0.4 \times 10^6\ nm^2/ms, & & & & \\
N_1 & = & 2300 \times 10^{-6}\ nm^{-2}, & & N_2 & = & 7500 \times 10^{-6}\ nm^{-2}, \\
N_2' & = & 750 \times 10^{-6}\ nm^{-2}, & & N_3 & = & 90 \times 10^{-6}\ nm^{-2}, \\
V_1 & = & 15.45 \times 10^3\ \#/ms, & & V_2 & = & 23.96 \times 10^3\ \#/ms, \\
V_3 & = & 1.98 \times 10^3\ \#/ms, & & & & \\
K_m^{(1)} & = & 12.044\ \#/nm^3, & & K_m^{(2)} & = & 10.8396\ \#/nm^3, \\
K_m^{(3)} & = & 16.8616\ \#/nm^3, & & & & \\
u_\infty & = & 0.0151\ \#/nm^3, & & & & \\
I & = & 30, & \Longrightarrow & \Delta r & = & 5\ nm, \\
J & = & 31, & \Longrightarrow & \Delta\theta & = & \pi/16, \\
K & = & 4, & \Longrightarrow & \Delta z & = & 5\ nm,
\end{array}
$$

Figure 12 shows the simulated dynamics of averaged glutamate concentration on the surface of the postsynaptic density ”PSD”. The dynamic changes with parameter ρ values: the larger is the astrocytic coverage of the synapse, the slower is the clearance of the cleft. Even in the case of high frequency stimulation ($100Hz$) and high values of ρ, the glutamate was cleared from the synaptic cleft between pulses. The complete coverage ($\rho = 1$) caused a drastic slowdown in synaptic clearance and accumulation of glutamate between high frequency pulses ($100Hz$). Let us note, however, that a full envelopment of a synapse by an astrocyte is very unlikely in reality [5]. We decided to further study EAATs' relative role on the glutamate dynamics. We simulated their blockage (by setting their turnover rates to 0) with no significant effect even in a high—coverage case ($\rho = 0.95$). According to our simulations, the rapid glutamate concentration rise and peak decay in the proximity of the postsynaptic receptors is achieved by the physical properties of the cleft itself, rather than active uptake of the transporters. It is unlikely that EAATs shape the time course of the glutamate on a scale of a single synapse despite the evidence collected on a scale of many synapses [23, 24, 25, 26, 27]. It appears that some other factors play a significant role in a prolongation of synaptic transmission upon blockage of glutamate transporters, e.g. it could be a previously mentioned spillover phenomenon.

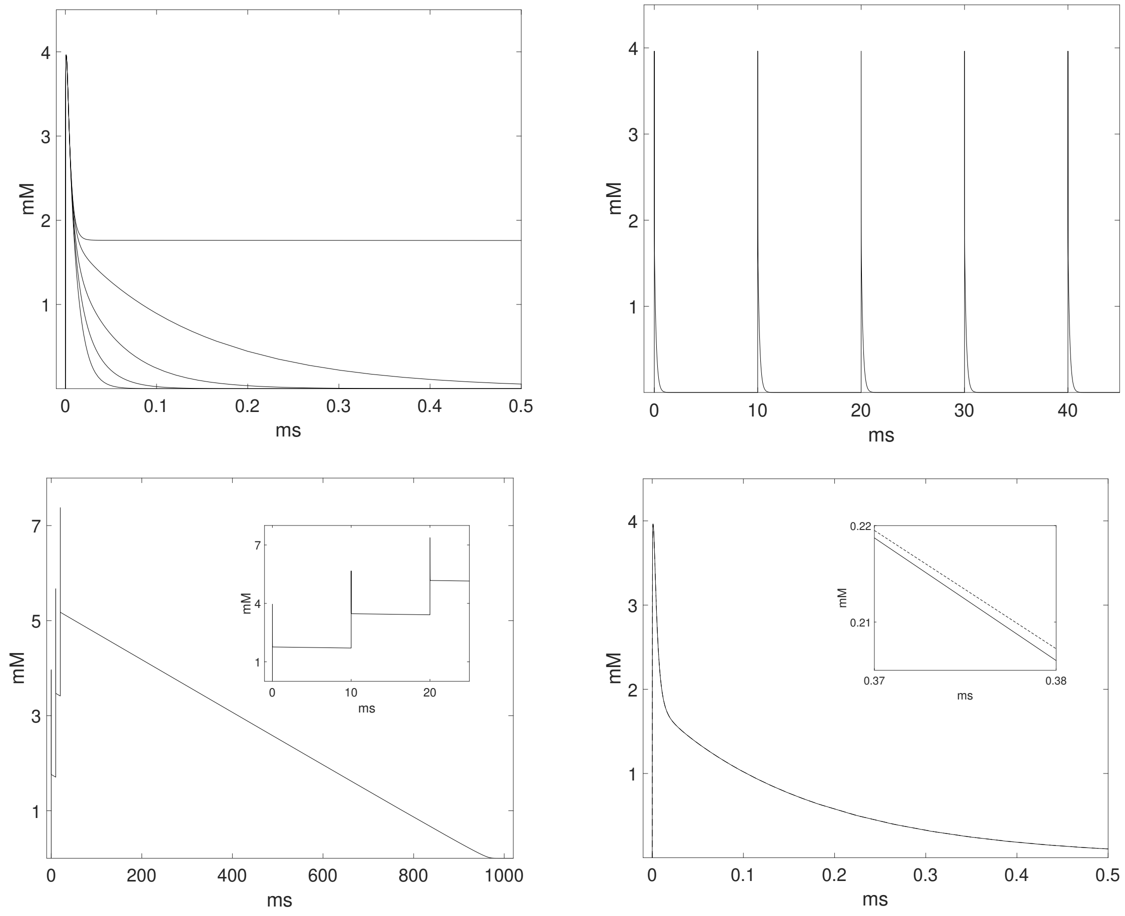

Figure 12: The simulated dynamics of the average glutamate concentrations on the surface of PSD. **Top left**. Results of a single stimulation with various proportion of astrocytic envelopment (fastest to slowest decays: $\rho = 0$, 0.3, 0.6, 0.9, 1). **Top right**. Series of 5 high-frequency vesicle releases ($100Hz$) with 95% complete astrocytic coverage. Glutamate is cleared from the synaptic cleft and does not accumulate between the pulses. **Bottom left**. Series of 3 at $100Hz$ in case $\rho = 1$. Glutamate accumulates between the pulses and requires around 1000 ms to clear from the synaptic cleft via facilitated transport by EAATs. The inset graph shows the first $25ms$ of the same plot that includes all three pulses. **Bottom right**. The comparison of a normal case (solid line) vs. transporters' blockage (dashed line). Even with high $\rho = 0.95$ the difference is insignificant: look at the inset graph, which is the zoomed portion of the same plot.

4 Glutamate receptors

One of the major subtypes of glutamate receptors on neurons is the N–methyl–D–aspartate receptor (NMDAR). NMDARs play critical roles in neural plasticity, development, learning, and memory. NMDAR was a target of multiple studies and several models were proposed to explain the dynamics of the processes in the ion channel protein. For common heterotetrameric NMDARs to signal, they must bind glutamate at each of the two NR2 subunits and coagonist (D–serine or glycine) at each of the two NR1-type subunits. [40]. In response to prolonged agonist pulses, NMDARs desensitize, a process in which the response amplitude decays over time. This desensitization is observed to increase in the presence of limiting coagonist [41, 42, 43]. This phenomenon could potentially be explained by different mechanisms. One possibility is that coagonist already bound to the NMDAR could experience a reduction in affinity following glutamate binding (”glycine—dependent desensitization” or, in our case, ”D–serine—dependent desensitization”). Alternatively, the effect of coagonist concentration on desensitization may not depend on agonist—coagonist site interactions [44, 45, 46, 47, 48, 49, 50]. The general chemical-kinetic model is shown in Figure 13. A number of studies that focus on that model, as well as some more complicated ones, have been published over the past two decades, e.g. [42, 51, 52]. However, all these models were overparameterized with respect to the available data. The authors tried to avoid this problem, for example, by fixing subsets of parameters while fitting the rest of them and then varied the fixed parameter values across numerous fitting procedures. Obviously this approach is not statistically correct and cannot be used to reliably estimate the parameters. If the statistically correct estimates for the parameters were found, it would answer the question about the nature of NMDAR desensitization.

Here we show how designing the experiments in accordance with model prediction resolves the issue of model overparameterization. We propose an algorithm with a series of experiments that can reliably estimate some of the parameters. Finally, as a proof of concept, we demonstrate the use of the algorithm on simulated data, which mimics the data from real experiments.

4.1 General model

The estimation of all parameters of the model depicted in Figure 13 would answer the question on the nature of NMDAR desensitization (e.g., if there is no desensitized state, then K_{13} will be negligibly small). However, as stated previously, such models are overparameterized with respect to the available data and cannot be used to produce meaningful estimations. One way to deal with the overparameterization problem is to decrease number of parameters under specific conditions, i.e., carefully chosen experiment designs, which allow an introduction of additional model assumptions.

We apply the patch clamp technique, described in details in Section 3.1. The experimenters have tight control of fluids that interact with NMDAR receptors. By raising or

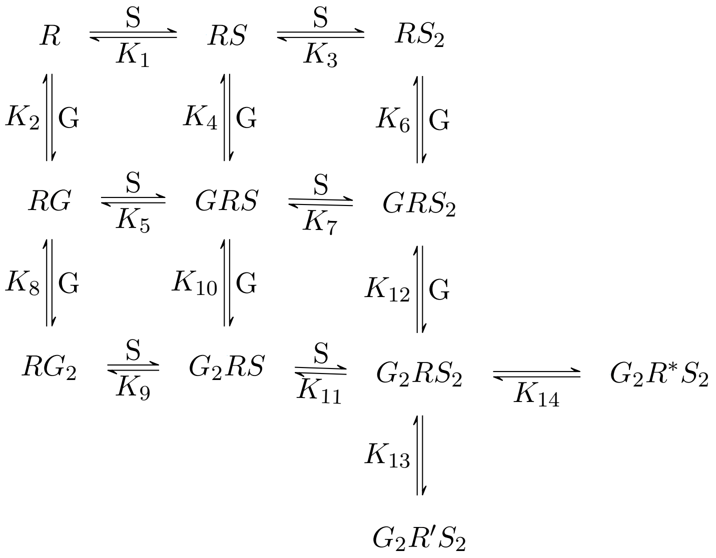

Figure 13: General model of NMDA receptor with two binding cites for L–glutamate and D–serine agonists. R denotes the receptor, S denotes D–serine, and G denotes L–glutamate. $G_2R'S_2$ is a long-lived nonconductive state and $G_2R^*S_2$ is a conductive state. Each K_i is an equilibrium constant for the corresponding reaction: $K_i = k_i^+/k_i^-$, where k_i^+ and k_i^- are forward and reverse reaction rate constants, respectively.

lowering the concentration of an NMDAR substrate, we effectively accelerate or decelerate the transitions involving this substrate. If we make the difference in concentrations high enough it will allow us to introduce a small parameter and apply the boundary function method (Appendix A.1) to derive reduced models with fewer parameters. It will also help us to get the insight on the nature of NMDAR desensitization. In all experiments one of the substrates is present continuously and the second one is applied in a single pulse manner. Thus, each experiment consists of three parts: before the pulse with only one agonist present, during the pulse with both substrates interacting with NMDAR, and after the pulse with again only one agonist present (the same as in the first part). We define two types of the experiments: Type 1 experiment is an experiment when D–serine is continuously present in the bath and L–glutamate is applied in a pulse-manner. The experiment of Type 1 becomes Type 2 experiment if D–serine and L–glutamate are switched. We also assume that the first part of the experiment is long enough for the system to reach a steady—state. Let us also note that, as in Section 3, the only feedback we are able to record is a current that corresponds to the flow of ions. This happens, i.e., the ion current is observed, only when the receptor is in a full—ligand conductive state, i.e., state $G_2R^*S_2$. Therefore, we do not observe any current during the first part of the experiment, we observe the current during the second part of the experiment due to the presence of both ligands in the bath, and we also observe some relaxation current during the third part of the experiment as one of the ligands dissociates and washes out from the bath. We do not need the whole system of differential equations to model the first part of the experiment as we cannot observe any feedback. We just have to find the corresponding steady–state, when one of the ligands is present, and use it as the initial conditions for the second part. For the sake of convenience, let us change the notation for the receptor's states in the model; see Figure 14.

$$\begin{tikzcd}
\alpha \arrow[r, leftrightharpoons, "\mathrm{S}", "K_1"'] \arrow[d, leftrightharpoons, "\mathrm{G}", "K_2"'] & \eta \arrow[r, leftrightharpoons, "\mathrm{S}", "K_3"'] \arrow[d, leftrightharpoons, "\mathrm{G}", "K_4"'] & x \arrow[d, leftrightharpoons, "\mathrm{G}", "K_6"'] & \\
\beta \arrow[r, leftrightharpoons, "\mathrm{S}", "K_5"'] \arrow[d, leftrightharpoons, "\mathrm{G}", "K_8"'] & \theta \arrow[r, leftrightharpoons, "\mathrm{S}", "K_7"'] \arrow[d, leftrightharpoons, "\mathrm{G}", "K_{10}"'] & y \arrow[d, leftrightharpoons, "\mathrm{G}", "K_{12}"'] & \\
\gamma \arrow[r, leftrightharpoons, "\mathrm{S}", "K_9"'] & \zeta \arrow[r, leftrightharpoons, "\mathrm{S}", "K_{11}"'] & z \arrow[r, leftrightharpoons, "K_{14}"'] \arrow[d, leftrightharpoons, "K_{13}"'] & z^* \\
 & & z' &
\end{tikzcd}$$

Figure 14: Here all variables (α, β, γ, χ, ϕ, ψ, x, y, z, z', z^*) represent concentrations of the corresponding compounds (see Figure 13); $K_i = k_i^+/k_i^-$, k_i^+ and k_i^- are the rate constants of forward (leftward and downward) and reverse (rightward and upward) reactions, respectively; S denotes the concentration of D–serine, G denotes L–glutamate.

For each state we define a corresponding variable, using the same notation, which is a fraction of all receptors observed to be in this state (e.g., the variable $x(t)$ is a fraction of all transporters in x state). The system has the following form:

$$
\begin{aligned}
\frac{d\alpha}{dt} &= -k_1^+\alpha S + k_1^-\eta - k_2^+\alpha G + k_2^-\beta,\\
\frac{d\eta}{dt} &= k_1^+\alpha S - k_1^-\eta - k_3^+\eta S + k_3^- x - k_4^+\eta G + k_4^-\theta,\\
\frac{dx}{dt} &= k_3^+\eta S - k_3^- x - k_6^+ xG + k_6^- y,\\
\frac{d\beta}{dt} &= k_2^+\alpha G - k_2^-\beta - k_5^+\beta S + k_5^-\theta - k_8^+\beta G + k_8^-\gamma,\\
\frac{d\theta}{dt} &= k_4^+\eta G - k_4^-\theta + k_5^+\beta S - k_5^-\theta - k_7^+\theta S + k_7^- y - k_{10}^+\theta G + k_{10}^-\zeta,\\
\frac{dy}{dt} &= k_6^+ xG - k_6^- y + k_7^+\theta S - k_7^- y - k_{12}^+ yG + k_{12}^- z,\\
\frac{d\gamma}{dt} &= k_8^+\beta G - k_8^-\gamma - k_9^+\gamma S + k_9^-\zeta,\\
\frac{d\zeta}{dt} &= k_9^+\gamma S - k_9^-\zeta + k_{10}^+\theta G - k_{10}^-\zeta - k_{11}^+\zeta S + k_{11}^- z,\\
\frac{dz}{dt} &= k_{11}^+\zeta S - k_{11}^- z + k_{12}^+ yG - k_{12}^- z - k_{13}^+ z + k_{13}^- z' - k_{14}^+ z + k_{14}^- z^*,\\
\frac{dz'}{dt} &= k_{13}^+ z - k_{13}^- z',\\
\frac{dz^*}{dt} &= k_{14}^+ z - k_{14}^- z^*.
\end{aligned}
\tag{25}
$$

This system describes parts two and three of all the experiments. The initial conditions for part three are the end states of part two. And, as mentioned earlier, the initial conditions for part two is the steady–state of part one. In Type 1 experiment, D–serine is always present. Thus, only the following states are present before the pulse: α, η, x. To find the steady–state $(\alpha_{ss}, \eta_{ss}, x_{ss})$, we have to solve:

$$
\begin{array}{rclcccccc}
0 &=& -k_1^+\alpha_{ss}S &+& k_1^-\eta_{ss}, & & & & \\
0 &=& k_1^+\alpha_{ss}S &-& k_1^-\eta_{ss} &-& k_3^+\eta_{ss}S &+& k_3^- x_{ss},\\
0 &=& k_3^+\eta_{ss}S &-& k_3^- x_{ss}. & & & &
\end{array}
$$

And thus,

$$\begin{aligned}\alpha_{ss} &= \frac{1}{1+K_1S+K_1K_3S^2},\\ \eta_{ss} &= \frac{K_1S}{1+K_1S+K_1K_3S^2},\\ x_{ss} &= \frac{K_1K_3S^2}{1+K_1S+K_1K_3S^2},\end{aligned} \tag{26}$$

where $K_i = k_i^+/k_i^-$ are equilibrium constants. Analogously, in Type 2 experiments, L–glutamate is always present, leading to the presence of the following states: α, β, γ. The corresponding steady–state $(\alpha_{ss}, \beta_{ss}, \gamma_{ss})$ is

$$\begin{aligned}\alpha_{ss} &= \frac{1}{1+K_2G+K_2K_8G^2},\\ \beta_{ss} &= \frac{K_2G}{1+K_2G+K_2K_8G^2},\\ \gamma_{ss} &= \frac{K_2K_8G^2}{1+K_2G+K_2K_8G^2}.\end{aligned} \tag{27}$$

Next, we focus on model reduction under different conditions.

4.2 Model reduction

This section includes he reduction of the model for various cases. It is split in subsections for the sake of convenience.

4.2.1 Model reduction during the pulse in the presence of high concentration of D–serine

Here we consider the reduction of the system (25) in the case of high concentration of D–serine (e.g., 10 mM) and low concentration of L–glutamate (e.g., 10 μM). We can introduce a small parameter ε:

$$S \gg 1, \quad \bar{S} = \varepsilon S, \quad \bar{S} = O(1), \quad 0 < \varepsilon \ll 1. \tag{28}$$

We now apply the Boundary Function Method (Appendix A.1). As before, we represent each state function as an asymptotic series with respect to the small parameter ε:

$$\begin{array}{rcrcl}\alpha(t) &=& \alpha_0(t) + \Pi_0\alpha(\tau) &+& \varepsilon\left(\alpha_1(t) + \Pi_1\alpha(\tau)\right) + \cdots,\\ \eta(t) &=& \eta_0(t) + \Pi_0\eta(\tau) &+& \varepsilon\left(\eta_1(t) + \Pi_1\eta(\tau)\right) + \cdots,\\ x(t) &=& x_0(t) + \Pi_0 x(\tau) &+& \varepsilon\left(x_1(t) + \Pi_1 x(\tau)\right) + \cdots,\\ && \vdots && \\ z^*(t) &=& z_0^*(t) + \Pi_0 z^*(\tau) &+& \varepsilon\left(z_1^*(t) + \Pi_1 z^*(\tau)\right) + \cdots,\end{array} \tag{29}$$

where $\tau = t/\varepsilon$ is a rescaled time variable. Let us remind that the functions which depend on t are called the regular functions, and the functions which depend on τ are called the boundary functions. Next, we substitute (28) and (29) into the system (25). Equating coefficients of like powers of ε, separately for the regular and the boundary layer functions, we obtain the reduced model problems for different terms of the asymptotic expansion (29). Using the standard asymptotic procedure (see [59]), for regular functions in the leading order approximation we obtain

$$\begin{aligned} 0 &= -k_1^+\bar{S}\alpha_0, \\ \frac{d\alpha_0}{dt} &= k_1^-\eta_0 - k_1^+\bar{S}\alpha_1 + k_2^+G\alpha_0 + k_2^-\beta_0, \end{aligned} \tag{30}$$

and

$$\begin{aligned} &\begin{array}{rclcrcl} \alpha_0(t) & \equiv & 0, & \qquad & \eta_0(t) & \equiv & 0, \\ \beta_0(t) & \equiv & 0, & & \theta_0(t) & \equiv & 0, \\ \gamma_0(t) & \equiv & 0, & & \zeta_0(t) & \equiv & 0, \end{array} \\ &\frac{dx_0}{dt} = k_3^+\bar{S}\eta_1 - k_3^-x_0 - k_6^+Gx_0 + k_6^-y_0, \\ &\frac{dy_0}{dt} = k_6^+Gx_0 - k_6^-y_0 + k_7^+\bar{S}\theta_1 - k_7^-y_0 - k_{12}^+Gy_0 + k_{12}^-z_0, \\ &\frac{dz_0}{dt} = k_{11}^+\bar{S}\zeta_1 - k_{11}^-z_0 + k_{12}^+Gy_0 - k_{12}^-z_0 - k_{13}^+z_0 + k_{13}^-z_0' - k_{14}^+z_0 + k_{14}^-z_0^*, \\ &\frac{dz_0'}{dt} = k_{13}^+z_0 - k_{13}^-z_0', \\ &\frac{dz_0^*}{dt} = k_{14}^+z_0 - k_{14}^-z_0^*. \end{aligned} \tag{31}$$

For the regular functions in the first order approximation we get:

$$\begin{array}{rcl} \alpha_1(t) & \equiv & 0, \\ \beta_1(t) & \equiv & 0, \\ \gamma_1(t) & \equiv & 0, \\ 0 & = & -k_3^+\bar{S}\eta_1 + k_3^-x_0, \\ 0 & = & -k_7^+\bar{S}\theta_1 + k_7^-y_0, \\ 0 & = & -k_{11}^+\bar{S}\zeta_1 + k_{11}^-z_0. \end{array} \tag{32}$$

Combining (30), (31), and (32), we obtain the closed system for regular functions in the leading order approximation:

$$
\begin{aligned}
&\alpha_0(t) \equiv 0, \qquad \eta_0(t) \equiv 0,\\
&\beta_0(t) \equiv 0, \qquad \theta_0(t) \equiv 0,\\
&\gamma_0(t) \equiv 0, \qquad \zeta_0(t) \equiv 0,\\
&\frac{dx_0}{dt} = -k_6^+ G x_0 + k_6^- y_0,\\
&\frac{dy_0}{dt} = k_6^+ G x_0 - k_6^- y_0 - k_{12}^+ G y_0 + k_{12}^- z_0,\\
&\frac{dz_0}{dt} = k_{12}^+ G y_0 - k_{12}^- z_0 - k_{13}^+ z_0 + k_{13}^- z_0' - k_{14}^+ z_0 + k_{14}^- z_0^*,\\
&\frac{dz_0'}{dt} = k_{13}^+ z_0 - k_{13}^- z_0',\\
&\frac{dz_0^*}{dt} = k_{14}^+ z_0 - k_{14}^- z_0^*.
\end{aligned}
\tag{33}
$$

For the boundary layer functions in the leading order approximation we obtain:

$$\begin{aligned}
\frac{d\Pi_0\alpha}{d\tau} &= -k_1^+\bar{S}\Pi_0\alpha,\\
\frac{d\Pi_0\eta}{d\tau} &= k_1^+\bar{S}\Pi_0\alpha - k_3^+\bar{S}\Pi_0\eta,\\
\frac{d\Pi_0x}{d\tau} &= k_3^+\bar{S}\Pi_0\eta,\\
\frac{d\Pi_0\beta}{d\tau} &= -k_5^+\bar{S}\Pi_0\beta,\\
\frac{d\Pi_0\theta}{d\tau} &= k_5^+\bar{S}\Pi_0\beta - k_7^+\bar{S}\Pi_0\theta,\\
\frac{d\Pi_0y}{d\tau} &= k_7^+\bar{S}\Pi_0\theta, \qquad (34)\\
\frac{d\Pi_0\gamma}{d\tau} &= -k_9^+\bar{S}\Pi_0\gamma,\\
\frac{d\Pi_0\zeta}{d\tau} &= k_9^+\bar{S}\Pi_0\gamma - k_{11}^+\bar{S}\Pi_0\zeta,\\
\frac{d\Pi_0z}{d\tau} &= k_{11}^+\bar{S}\Pi_0\zeta,\\
\frac{d\Pi_0z'}{d\tau} &= 0,\\
\frac{d\Pi_0z^*}{d\tau} &= 0.
\end{aligned}$$

The solutions for (34) can easily be found analytically. However, we need initial conditions to solve both systems, which depend on the type of the experiment.

We start with Type 1 experiment. The initial conditions for Type 1 experiment at the start of the pulse for the full model (25) are (26) for α, η, x, and zeros for all the other states. Taking into account (28), we can expand the initial conditions in powers of ε using Taylor's series. We get:

$$\alpha(0) = \alpha_{ss} = 0 + 0\cdot\varepsilon + O(\varepsilon^2),$$

$$\eta(0) = \eta_{ss} = 0 + \frac{\varepsilon}{K_3\bar{S}} + O(\varepsilon^2),$$

$$x(0) = x_{ss} = 1 - \frac{\varepsilon}{K_3\bar{S}} + O(\varepsilon^2).$$

Therefore,

$$\begin{array}{rcccl}
\alpha_0(0)+\Pi_0\alpha(0) &=& \Pi_0\alpha(0) &=& 0,\\
\eta_0(0)+\Pi_0\eta(0) &=& \Pi_0\eta(0) &=& 0,\\
x_0(0)+\Pi_0x(0) &=& 1, && \\
\beta_0(0)+\Pi_0\beta(0) &=& \Pi_0\beta(0) &=& 0,\\
\theta_0(0)+\Pi_0\theta(0) &=& \Pi_0\theta(0) &=& 0,\\
y_0(0)+\Pi_0y(0) &=& 0, && \\
\gamma_0(0)+\Pi_0\gamma(0) &=& \Pi_0\gamma(0) &=& 0,\\
\zeta_0(0)+\Pi_0\zeta(0) &=& \Pi_0\zeta(0) &=& 0,\\
z_0(0)+\Pi_0z(0) &=& 0, && \\
z_0'(0)+\Pi_0z'(0) &=& 0, && \\
z_0^*(0)+\Pi_0z^*(0) &=& 0. &&
\end{array}$$

From (34) and using the fact that all boundary functions must tend to zero as τ goes to infinity, we find that all the boundary functions become equal to zero in this case. Moreover, we get the initial conditions for the system of regular functions in the leading order approximation (33):

$$\begin{array}{rcl}
x_0(0) &=& 1,\\
y_0(0) &=& 0,\\
z_0(0) &=& 0,\\
z_0'(0) &=& 0,\\
z_0^*(0) &=& 0.
\end{array} \qquad (35)$$

Together, the system (33) with the initial conditions (35) and zero boundary functions approximate the dynamics of the full system (25) according to the expansion (29) in Type 1 experiment with high concentration of D–serine and low concentration of L–glutamate, if the system is allowed to reach a steady–state before the pulse of glutamate is applied.

The initial conditions for Type 2 experiment at the start of the pulse for the full model (25) are (27) for α, β, γ, and zeros for all the other states. There is no small parameter in (27), therefore, just using (29), we get

$$\begin{array}{rcccl}
\alpha_0(0)+\Pi_0\alpha(0) &=& \Pi_0\alpha(0) &=& \alpha_{ss},\\
\eta_0(0)+\Pi_0\eta(0) &=& \Pi_0\eta(0) &=& 0,\\
x_0(0)+\Pi_0x(0) &=& 0, && \\
\beta_0(0)+\Pi_0\beta(0) &=& \Pi_0\beta(0) &=& \beta_{ss},\\
\theta_0(0)+\Pi_0\theta(0) &=& \Pi_0\theta(0) &=& 0,\\
y_0(0)+\Pi_0y(0) &=& 0, && \\
\gamma_0(0)+\Pi_0\gamma(0) &=& \Pi_0\gamma(0) &=& \gamma_{ss},\\
\zeta_0(0)+\Pi_0\zeta(0) &=& \Pi_0\zeta(0) &=& 0,\\
z_0(0)+\Pi_0z(0) &=& 0, && \\
z_0'(0)+\Pi_0z'(0) &=& 0, && \\
z_0^*(0)+\Pi_0z^*(0) &=& 0. &&
\end{array}$$

From (34) and using the fact that all boundary functions must tend to zero as τ goes to infinity, we get that

$$
\begin{aligned}
\Pi_0\alpha(\tau) &= \alpha_{ss}e^{-k_1^+\bar{S}\tau}, \\
\Pi_0\eta(\tau) &= \frac{k_1^+\alpha_{ss}}{k_3^+-k_1^+}\left(e^{-k_1^+\bar{S}\tau}-e^{-k_3^+\bar{S}\tau}\right), \\
\Pi_0x(\tau) &= \frac{k_1^+k_3^+\alpha_{ss}}{k_3^+-k_1^+}\left(\frac{1}{k_3^+}e^{-k_3^+\bar{S}\tau}-\frac{1}{k_1^+}e^{-k_1^+\bar{S}\tau}\right), \\
\Pi_0\beta(\tau) &= \beta_{ss}e^{-k_5^+\bar{S}\tau}, \\
\Pi_0\theta(\tau) &= \frac{k_5^+\beta_{ss}}{k_7^+-k_5^+}\left(e^{-k_5^+\bar{S}\tau}-e^{-k_7^+\bar{S}\tau}\right), \\
\Pi_0y(\tau) &= \frac{k_5^+k_7^+\beta_{ss}}{k_7^+-k_5^+}\left(\frac{1}{k_7^+}e^{-k_7^+\bar{S}\tau}-\frac{1}{k_5^+}e^{-k_5^+\bar{S}\tau}\right), \\
\Pi_0\gamma(\tau) &= \gamma_{ss}e^{-k_9^+\bar{S}\tau}, \\
\Pi_0\zeta(\tau) &= \frac{k_9^+\gamma_{ss}}{k_{11}^+-k_9^+}\left(e^{-k_9^+\bar{S}\tau}-e^{-k_{11}^+\bar{S}\tau}\right), \\
\Pi_0z(\tau) &= \frac{k_9^+k_{11}^+\gamma_{ss}}{k_{11}^+-k_9^+}\left(\frac{1}{k_{11}^+}e^{-k_{11}^+\bar{S}\tau}-\frac{1}{k_9^+}e^{-k_9^+\bar{S}\tau}\right), \\
\Pi_0z'(\tau) &\equiv 0, \\
\Pi_0z^*(\tau) &\equiv 0.
\end{aligned}
\tag{36}
$$

We also get the following initial conditions for the system of regular functions in leading order approximation (33):

$$
\begin{aligned}
x_0(0) &= \alpha_{ss}, \\
y_0(0) &= \beta_{ss}, \\
z_0(0) &= \gamma_{ss}, \\
z_0'(0) &= 0, \\
z_0^*(0) &= 0.
\end{aligned}
\tag{37}
$$

Together, the system (33) with the initial conditions (37), and the boundary functions (36), approximate the dynamics of the full system (25) according to the expansion (29) in Type 2 experiment with high concentration of D–serine and low concentration of L–glutamate, if it is allowed to reach a steady–state before the pulse of serine is applied.

4.2.2 Model reduction during the pulse in the presence of high concentration of L–glutamate

We now consider a mirrored case of the reduction of system (25) in the case of high concentration of L–glutamate (e.g., 10 mM) and low concentration of D–serine (e.g., 10 μM). Now we introduce a small parameter ε:

$$G \gg 1, \quad \bar{G} = \varepsilon G, \quad \bar{G} = O(1), \quad 0 < \varepsilon \ll 1. \tag{38}$$

As before, we apply the Boundary Function Method (Appendix A.1) and represent each state function of the system as an asymptotic series (29). Following the same procedure of equating coefficients of like powers of ε separately for the regular and the boundary layer functions, we obtain the systems analogous to (33) and (34) from the previous case:

$$\begin{array}{rclcrcl}
\alpha_0(t) & \equiv & 0, & \qquad & \eta_0(t) & \equiv & 0, \\
\beta_0(t) & \equiv & 0, & & \theta_0(t) & \equiv & 0, \\
x_0(t) & \equiv & 0, & & y_0(t) & \equiv & 0,
\end{array}$$

$$\begin{aligned}
\frac{d\gamma_0}{dt} &= -k_9^+ S\gamma_0 + k_9^- \zeta_0, \\
\frac{d\zeta_0}{dt} &= k_9^+ S\gamma_0 - k_9^- \zeta_0 - k_{11}^+ S\zeta_0 + k_{11}^- z_0, \\
\frac{dz_0}{dt} &= k_{11}^+ S\zeta_0 - k_{11}^- z_0 - k_{13}^+ z_0 + k_{13}^- z_0' - k_{14}^+ z_0 + k_{14}^- z_0^*, \\
\frac{dz_0'}{dt} &= k_{13}^+ z_0 - k_{13}^- z_0', \\
\frac{dz_0^*}{dt} &= k_{14}^+ z_0 - k_{14}^- z_0^*,
\end{aligned} \tag{39}$$

and

$$\begin{aligned}
\frac{d\Pi_0\alpha}{d\tau} &= -k_2^+\bar{G}\Pi_0\alpha,\\
\frac{d\Pi_0\eta}{d\tau} &= -k_4^+\bar{G}\Pi_0\eta,\\
\frac{d\Pi_0 x}{d\tau} &= -k_6^+\bar{G}\Pi_0 x,\\
\frac{d\Pi_0\beta}{d\tau} &= k_2^+\bar{G}\Pi_0\alpha - k_8^+\bar{G}\Pi_0\beta,\\
\frac{d\Pi_0\theta}{d\tau} &= k_4^+\bar{G}\Pi_0\eta - k_{10}^+\bar{G}\Pi_0\theta,\\
\frac{d\Pi_0 y}{d\tau} &= k_6^+\bar{G}\Pi_0 x - k_{12}^+\bar{G}\Pi_0 y,\\
\frac{d\Pi_0\gamma}{d\tau} &= k_8^+\bar{G}\Pi_0\beta,\\
\frac{d\Pi_0\zeta}{d\tau} &= k_{10}^+\bar{G}\Pi_0\theta,\\
\frac{d\Pi_0 z}{d\tau} &= k_{12}^+\bar{G}\Pi_0 y,\\
\frac{d\Pi_0 z'}{d\tau} &= 0,\\
\frac{d\Pi_0 z^*}{d\tau} &= 0.
\end{aligned} \tag{40}$$

As before, we consider two types of the experiment that will define initial conditions for both systems (39) and (40).

The initial conditions for Type 1 experiment at the start of the pulse for the full model (25) are (26) for α, η, x, and zeros for all other states. There is no small parameter in (26), therefore

$$\begin{array}{rcccc}
\alpha_0(0)+\Pi_0\alpha(0) & = & \Pi_0\alpha(0) & = & \alpha_{ss},\\
\eta_0(0)+\Pi_0\eta(0) & = & \Pi_0\eta(0) & = & \eta_{ss},\\
x_0(0)+\Pi_0 x(0) & = & \Pi_0 x(0) & = & x_{ss},\\
\beta_0(0)+\Pi_0\beta(0) & = & \Pi_0\beta(0) & = & 0,\\
\theta_0(0)+\Pi_0\theta(0) & = & \Pi_0\theta(0) & = & 0,\\
y_0(0)+\Pi_0 y(0) & = & \Pi_0 y(0) & = & 0,\\
\gamma_0(0)+\Pi_0\gamma(0) & = & 0, & & \\
\zeta_0(0)+\Pi_0\zeta(0) & = & 0, & & \\
z_0(0)+\Pi_0 z(0) & = & 0, & & \\
z'_0(0)+\Pi_0 z'(0) & = & 0, & & \\
z^*_0(0)+\Pi_0 z^*(0) & = & 0. & &
\end{array}$$

Using (40) and the fact that all boundary functions must rend to zero as τ goes to infinity, we get

$$\begin{aligned}
\Pi_0\alpha(\tau)\ &= \alpha_{ss}e^{-k_2^+\bar{G}\tau},\\
\Pi_0\eta(\tau)\ &= \eta_{ss}e^{-k_4^+\bar{G}\tau},\\
\Pi_0x(\tau)\ &= x_{ss}e^{-k_6^+\bar{G}\tau},\\
\Pi_0\beta(\tau)\ &= \frac{k_2^+\alpha_{ss}}{k_8^+-k_2^+}\left(e^{-k_2^+\bar{G}\tau}-e^{-k_8^+\bar{G}\tau}\right),\\
\Pi_0\theta(\tau)\ &= \frac{k_4^+\eta_{ss}}{k_{10}^+-k_4^+}\left(e^{-k_4^+\bar{G}\tau}-e^{-k_{10}^+\bar{G}\tau}\right),\\
\Pi_0y(\tau)\ &= \frac{k_6^+x_{ss}}{k_{12}^+-k_6^+}\left(e^{-k_6^+\bar{G}\tau}-e^{-k_{12}^+\bar{G}\tau}\right),\\
\Pi_0\gamma(\tau)\ &= \frac{k_2^+k_8^+\alpha_{ss}}{k_8^+-k_2^+}\left(\frac{1}{k_8^+}e^{-k_8^+\bar{G}\tau}-\frac{1}{k_2^+}e^{-k_2^+\bar{G}\tau}\right),\\
\Pi_0\zeta(\tau)\ &= \frac{k_4^+k_{10}^+\eta_{ss}}{k_{10}^+-k_4^+}\left(\frac{1}{k_{10}^+}e^{-k_{10}^+\bar{G}\tau}-\frac{1}{k_4^+}e^{-k_4^+\bar{G}\tau}\right),\\
\Pi_0z(\tau)\ &= \frac{k_6^+k_{12}^+x_{ss}}{k_{12}^+-k_6^+}\left(\frac{1}{k_{12}^+}e^{-k_{12}^+\bar{G}\tau}-\frac{1}{k_6^+}e^{-k_6^+\bar{G}\tau}\right),\\
\Pi_0z'(\tau)&\equiv 0,\\
\Pi_0z^*(\tau)&\equiv 0.
\end{aligned}\tag{41}$$

We also get initial conditions for the system of regular functions in the leading order approximation (39):

$$\begin{aligned}
\gamma_0(0) &= \alpha_{ss},\\
\zeta_0(0) &= \eta_{ss},\\
z_0(0) &= x_{ss},\\
z_0'(0) &= 0,\\
z_0^*(0) &= 0.
\end{aligned}\tag{42}$$

Together, the system (39) with the initial conditions (42), and the boundary functions (41), approximate the dynamics of the full system (25) according to the expansion (29) in Type 1 experiment with high concentration of L–glutamate and low concentration of D–serine, if it is allowed to reach a steady–state before the pulse of glutamate is applied.

The initial conditions for Type 2 experiment at the start of the pulse for the full model (25) are (27) for α, β, γ, and zeros for all other states. Taking into account (38), we can expand the initial conditions in powers of ε using Taylor's series. We get:

$$\begin{aligned} \alpha(0) &= \alpha_{ss} = 0 + 0 \cdot \varepsilon + O(\varepsilon^2), \\ \beta(0) &= \beta_{ss} = 0 + \frac{\varepsilon}{K_8 \bar{G}} + O(\varepsilon^2), \\ \gamma(0) &= \gamma_{ss} = 1 - \frac{\varepsilon}{K_8 \bar{G}} + O(\varepsilon^2). \end{aligned}$$

As before, for the leading order approximation functions we get:

$$\begin{aligned} \alpha_0(0) + \Pi_0\alpha(0) &= \Pi_0\alpha(0) = 0, \\ \eta_0(0) + \Pi_0\eta(0) &= \Pi_0\eta(0) = 0, \\ x_0(0) + \Pi_0 x(0) &= \Pi_0 x(0) = 0, \\ \beta_0(0) + \Pi_0\beta(0) &= \Pi_0\beta(0) = 0, \\ \theta_0(0) + \Pi_0\theta(0) &= \Pi_0\theta(0) = 0, \\ y_0(0) + \Pi_0 y(0) &= \Pi_0 y(0) = 0, \\ \gamma_0(0) + \Pi_0\gamma(0) &= 1, \\ \zeta_0(0) + \Pi_0\zeta(0) &= 0, \\ z_0(0) + \Pi_0 z(0) &= 0, \\ z_0'(0) + \Pi_0 z'(0) &= 0, \\ z_0^*(0) + \Pi_0 z^*(0) &= 0. \end{aligned}$$

Using (40) and the fact that all boundary functions must tend to zero as τ goes to infinity, we find that all the boundary functions become equal to zero in this case. Moreover, we get the initial conditions for the system of regular functions in the leading order approximation (39):

$$\begin{aligned} \gamma_0(0) &= 1, \\ \zeta_0(0) &= 0, \\ z_0(0) &= 0, \\ z_0'(0) &= 0, \\ z_0^*(0) &= 0. \end{aligned} \tag{43}$$

Together, the system (39) with the initial conditions (43), and zero boundary functions approximate the dynamics of the full system (25) according to the expansion (29) in Type 2 experiment with high concentration of L–glutamate and low concentration of D–serine, if it is allowed to reach a steady–state before the pulse of serine is applied.

4.2.3 Model reduction during the pulse in the presence of high concentrations of D–serine and L–glutamate

Suppose that the concentrations of both D–serine and L–glutamate are high. Then we can introduce a small parameter ε:

$$S \gg 1, \quad G \gg 1, \quad \bar{S} = \varepsilon S, \quad \bar{G} = \varepsilon G, \quad \bar{S} = O(1), \quad \bar{G} = O(1), \quad 0 < \varepsilon \ll 1. \tag{44}$$

We apply the Boundary Function Method (Appendix A.1) and represent each state function of the system as an asymptotic series (29). Following the same procedure of equating coefficients of like powers of ε separately for regular and the boundary layer functions, we obtain the following systems

$$\begin{array}{rclcrcl}
\alpha_0(t) & \equiv & 0, & \qquad & \eta_0(t) & \equiv & 0, \\
\beta_0(t) & \equiv & 0, & & \theta_0(t) & \equiv & 0, \\
x_0(t) & \equiv & 0, & & y_0(t) & \equiv & 0, \\
\gamma_0(t) & \equiv & 0, & & \zeta_0(t) & \equiv & 0,
\end{array}$$

$$\begin{aligned}
\frac{dz_0}{dt} &= -k_{13}^{+} z_0 + k_{13}^{-} z_0' - k_{14}^{+} z_0 + k_{14}^{-} z_0^*, \\
\frac{dz_0'}{dt} &= k_{13}^{+} z_0 - k_{13}^{-} z_0', \\
\frac{dz_0^*}{dt} &= k_{14}^{+} z_0 - k_{14}^{-} z_0^*,
\end{aligned} \tag{45}$$

and

$$\begin{aligned}
\frac{d\Pi_0\alpha}{d\tau} &= -k_1^{+}\bar{S}\Pi_0\alpha - k_2^{+}\bar{G}\Pi_0\alpha, \\
\frac{d\Pi_0\eta}{d\tau} &= k_1^{+}\bar{S}\Pi_0\alpha - k_3^{+}\bar{S}\Pi_0\eta - k_4^{+}\bar{G}\Pi_0\eta, \\
\frac{d\Pi_0 x}{d\tau} &= k_3^{+}\bar{S}\Pi_0\eta - k_6^{+}\bar{G}\Pi_0 x, \\
\frac{d\Pi_0\beta}{d\tau} &= k_2^{+}\bar{G}\Pi_0\alpha - k_5^{+}\bar{S}\Pi_0\beta - k_8^{+}\bar{G}\Pi_0\beta, \\
\frac{d\Pi_0\theta}{d\tau} &= k_4^{+}\bar{G}\Pi_0\eta + k_5^{+}\bar{S}\Pi_0\beta - k_7^{+}\bar{S}\Pi_0\theta - k_{10}^{+}\bar{G}\Pi_0\theta, \\
\frac{d\Pi_0 y}{d\tau} &= k_6^{+}\bar{G}\Pi_0 x + k_7^{+}\bar{S}\Pi_0\theta - k_{12}^{+}\bar{G}\Pi_0 y, \\
\frac{d\Pi_0\gamma}{d\tau} &= k_8^{+}\bar{G}\Pi_0\beta - k_9^{+}\bar{S}\Pi_0\gamma, \\
\frac{d\Pi_0\zeta}{d\tau} &= k_9^{+}\bar{S}\Pi_0\gamma + k_{10}^{+}\bar{G}\Pi_0\theta - k_{11}^{+}\bar{S}\Pi_0\zeta, \\
\frac{d\Pi_0 z}{d\tau} &= k_{11}^{+}\bar{S}\Pi_0\zeta + k_{12}^{+}\bar{G}\Pi_0 y, \\
\frac{d\Pi_0 z'}{d\tau} &= 0, \\
\frac{d\Pi_0 z^*}{d\tau} &= 0.
\end{aligned} \tag{46}$$

As before, we will consider two types of the experiment which will define initial conditions for both systems (45) and (46).

The initial conditions for Type 1 experiment at the start of the pulse for full model (25) are (26) for α, η, x, and zeros for all other states. Using (44) we can write the expansion of the initial conditions in powers of ε and get:

$$\begin{array}{rcccl}
\alpha_0(0) + \Pi_0\alpha(0) & = & \Pi_0\alpha(0) & = & 0, \\
\eta_0(0) + \Pi_0\eta(0) & = & \Pi_0\eta(0) & = & 0, \\
x_0(0) + \Pi_0 x(0) & = & \Pi_0 x(0) & = & 1, \\
\beta_0(0) + \Pi_0\beta(0) & = & \Pi_0\beta(0) & = & 0, \\
\theta_0(0) + \Pi_0\theta(0) & = & \Pi_0\theta(0) & = & 0, \\
y_0(0) + \Pi_0 y(0) & = & \Pi_0 y(0) & = & 0, \\
\gamma_0(0) + \Pi_0\gamma(0) & = & \Pi_0\gamma(0) & = & 0, \\
\zeta_0(0) + \Pi_0\zeta(0) & = & \Pi_0\zeta(0) & = & 0, \\
z_0(0) + \Pi_0 z(0) & = & 0, & & \\
z_0'(0) + \Pi_0 z'(0) & = & 0, & & \\
z_0^*(0) + \Pi_0 z^*(0) & = & 0. & &
\end{array}$$

Then from (46) we get:

$$\begin{aligned}
\Pi_0\alpha(\tau) &\equiv 0,\\
\Pi_0\eta(\tau) &\equiv 0,\\
\Pi_0 x(\tau) &= e^{-k_6^+\bar{G}\tau},\\
\Pi_0\beta(\tau) &\equiv 0,\\
\Pi_0\theta(\tau) &\equiv 0,\\
\Pi_0 y(\tau) &= \frac{k_6^+}{k_{12}^+ - k_6^+}\left(e^{-k_6^+\bar{G}\tau} - e^{-k_{12}^+\bar{G}\tau}\right), && (47)\\
\Pi_0\gamma(\tau) &\equiv 0,\\
\Pi_0\zeta(\tau) &\equiv 0,\\
\Pi_0 z(\tau) &= \frac{k_6^+ k_{12}^+}{k_{12}^+ - k_6^+}\left(\frac{1}{k_{12}^+}e^{-k_{12}^+\bar{G}\tau} - \frac{1}{k_6^+}e^{-k_6^+\bar{G}\tau}\right),\\
\Pi_0 z'(\tau) &\equiv 0,\\
\Pi_0 z^*(\tau) &\equiv 0.
\end{aligned}$$

We also get initial conditions for the system describing the regular functions in leading order approximation (45):

$$\begin{aligned}
z_0(0) &= 1,\\
z_0'(0) &= 0, && (48)\\
z_0^*(0) &= 0.
\end{aligned}$$

Together, the system (45) with the initial conditions (48), and the boundary functions (47), approximate the dynamics of the full system (25) according to the expansion (29) in Type 1 experiment with high concentration of both D–serine and L–glutamate, if it is allowed to reach a steady–state before the pulse of glutamate is applied.

The initial conditions for Type 2 experiment at the start of the pulse for the full model (25) are (27) for α, β, γ, and zeros for all other states. Using (44) we can write the expansion of the initial conditions in powers of ε and get:

$$\begin{array}{rcrcl}
\alpha_0(0)+\Pi_0\alpha(0) &=& \Pi_0\alpha(0) &=& 0,\\
\eta_0(0)+\Pi_0\eta(0) &=& \Pi_0\eta(0) &=& 0,\\
x_0(0)+\Pi_0x(0) &=& \Pi_0x(0) &=& 0,\\
\beta_0(0)+\Pi_0\beta(0) &=& \Pi_0\beta(0) &=& 0,\\
\theta_0(0)+\Pi_0\theta(0) &=& \Pi_0\theta(0) &=& 0,\\
y_0(0)+\Pi_0y(0) &=& \Pi_0y(0) &=& 0,\\
\gamma_0(0)+\Pi_0\gamma(0) &=& \Pi_0\gamma(0) &=& 1,\\
\zeta_0(0)+\Pi_0\zeta(0) &=& \Pi_0\zeta(0) &=& 0,\\
z_0(0)+\Pi_0z(0) &=& 0, && \\
z'_0(0)+\Pi_0z'(0) &=& 0, && \\
z^*_0(0)+\Pi_0z^*(0) &=& 0. &&
\end{array}$$

Then from (46) we get

$$\begin{array}{rcl}
\Pi_0\alpha(\tau) &\equiv& 0, \qquad \Pi_0\eta(\tau) \equiv 0,\\
\Pi_0\beta(\tau) &\equiv& 0, \qquad \Pi_0\theta(\tau) \equiv 0,\\
\Pi_0x(\tau) &\equiv& 0, \qquad \Pi_0y(\tau) \equiv 0,
\end{array}$$

$$\begin{aligned}
\Pi_0\gamma(\tau) &= e^{-k_9^+\bar{S}\tau},\\
\Pi_0\zeta(\tau) &= \frac{k_9^+}{k_{11}^+-k_9^+}\left(e^{-k_9^+\bar{S}\tau}-e^{-k_{11}^+\bar{S}\tau}\right),\\
\Pi_0z(\tau) &= \frac{k_9^+k_{11}^+}{k_{11}^+-k_9^+}\left(\frac{1}{k_{11}^+}e^{-k_{11}^+\bar{S}\tau}-\frac{1}{k_9^+}e^{-k_9^+\bar{S}\tau}\right),\\
\Pi_0z'(\tau) &\equiv 0,\\
\Pi_0z^*(\tau) &\equiv 0,
\end{aligned} \tag{49}$$

We also obtain the initial conditions for the system of regular functions in the leading order approximation (45):

$$\begin{aligned}
z_0(0) &= 1,\\
z'_0(0) &= 0,\\
z^*_0(0) &= 0.
\end{aligned} \tag{50}$$

Together, the system (45) with the initial conditions (48), and the boundary functions (49) approximate the dynamics of the full system (25) according to the expansion (29) in Type 2 experiment with high concentration of both D–serine and L–glutamate, if it is allowed to reach a steady–state before the pulse of serine is applied.

4.2.4 Model reduction after the pulse for all cases

After the pulse ends in each experiment, the concentrations of substances (and our assumptions) change, which can lead to a need for a new model. Here, we will consider all 6 cases discussed above. Let us note that the initial conditions for new models after the pulse are approximated by the asymptotic expansion (29) and depends on the final states of corresponding reduced models during the pulse. We assume that the second part of experiment is long enough so that the fast decaying boundary functions reach negligibly small values by the end of the pulse.

We start with going over the easiest cases when the model does not actually change. In Type 1 experiment (D–serine is continuously present in the bath and L–glutamate is applied in a pulse manner) when the concentration of D–serine is high and the concentration of L–glutamate is low, the conditions after the pulse do not change: D–serine continues to be present at high concentration, while L–glutamate concentration switches from low to zero. Therefore, the same reduced model can be used: (33) for the regular leading order approximation functions and (34) for the boundary leading order approximation functions. Moreover only states x, y, z, z', and z^* are nonzero in the leading order approximation, therefore, all the boundary functions are equal to zero. Thus, the third part of Type 1 experiment in the presence of high concentration of D–serine is approximated by system (33) (with $G = 0$ and initial conditions being the end state of the previous stage of experiment) and zero boundary functions according to the expansion (29).

Analogously, In Type 2 experiment (L–glutamate is continuously present in the bath and D–serine is applied in a pulse manner) when the concentration of L–glutamate is high and the concentration of D–serine is low, the conditions after the pulse do not change: L–glutamate continues to be present at high concentration, while D–serine switches from low to zero. Following the same logic, the third part of Type 2 experiment in the presence of high concentration of L–glutamate is approximated by system (39) (with $S = 0$ and initial conditions corresponding to the end state of the previous stage of experiment) and zero boundary functions according to the expansion (29).

We consider now more complicated cases. In Type 1 experiment when the concentration of L–glutamate is high and the concentration of D–serine is low, the conditions after the pulse change: while D–serine continues to be present at low concentrations, L–glutamate concentration switches from high to zero. The reduction is not possible in this case and we must use the full system (25) with $G = 0$ to describe the washout portion of this experiment. Again, the initial conditions of the system are the end state of the previous stage of experiment.

Analogously, in Type 2 experiment when the concentration of D–serine is high and the concentration of L–glutamate is low, the reduction is not possible. Thus, the full system (25) with $S = 0$ must be used to describe the washout portion of this experiment.

Finally, following the same logic, when concentrations of both D–serine and L–glutamate are high we should use system (33) with $G = 0$ to describe the third part of Type 1 experiment and system (39) with $S = 0$ to describe the third part of Type 2 experiment.

4.3 Model application

By varying the conditions of the experiments we can change the number of parameters in the corresponding reduced model. Performing the experiments in a correct order allows us to estimate parameters in a step–wise manner, using the estimates obtained during the previous step in the consecutive step. That is, the following sequence of experiments and fitting to the corresponding models allows the estimation of some parameters of the model depicted in Figures 13, 14:

1. Type 1 and Type 2 experiments with high concentrations of both substrates. The reduced model describing the second part of the experiment contains parameters $k_{13}^{\pm}$ and $k_{14}^{\pm}$.

2. Type 1 experiment with saturating D–serine and low concentration of L–glutamate; third part of Type 1 experiment with saturating concentrations of both ligands. The reduced model contains additional parameters $k_6^{\pm}$ and $k_{12}^{\pm}$.

3. Type 2 experiment with saturating L–glutamate and low concentration of D–serine; third part of Type 2 experiment with saturating concentrations of both ligands. The reduced model contains additional parameters $k_9^{\pm}$ and $k_{11}^{\pm}$.

4. Type 1 experiment with saturating L–glutamate and low concentration of D–serine. The reduced model is identical to the one in step 3 (except boundary functions that do not enter into the formula of the recorded current, which is just proportional to the state z^*). The initial conditions contain additional parameters: $K_1 = k_1^+/k_1^-$ and $K_3 = k_3^+/k_3^-$.

5. Type 2 experiment with saturating D–serine and low concentration of L–glutamate. The reduced model is identical to the on in step 2 (with the same remark as in step 4). However, initial conditions contain additional parameters: $K_2 = k_2^+/k_2^-$ and $K_8 = k_8^+/k_8^-$.

The approach described above guarantees that we do not have overparameterized models as the number of parameters at each step is sufficiently low and may be reliably estimated. Let us also note that depending on the data the estimation of the parameters K_1, K_2, K_3, and K_8 may be numerically unstable.

We note that we do not have the experimental data to perform the estimation procedure and answer the question about the nature of NMDAR desensitization. Here we show the proof of concept and illustrate the described process using the simulated data. As we mentioned previously, there are several studies that tried to overcome the overparameterization by performing statistically incorrect procedures. The results of these procedures

Parameter	True value	Estimated value
k_{13}^{+}	3.68	3.65 ± 0.02
k_{13}^{-}	3.00	3.00 ± 0.01
k_{14}^{+}	83.80	81.93 ± 0.63
k_{14}^{-}	83.80	83.45 ± 0.19
k_{6}^{+}	4.00	4.00 ± 0.01
k_{6}^{-}	0.97	1.42 ± 0.23
k_{12}^{+}	4.00	4.04 ± 0.02
k_{12}^{-}	8.25	8.17 ± 0.00
k_{9}^{+}	1.70	1.70 ± 0.01
k_{9}^{-}	2.35	2.36 ± 0.17
k_{11}^{+}	1.70	1.70 ± 0.02
k_{11}^{-}	19.90	19.73 ± 0.02
K_1	0.7234	0.5537 ± 0.08
K_3	0.7234	0.7720 ± 0.03
K_2	4.1237	15.74 ± 60.78
K_8	4.1237	3.82 ± 0.26

Table 1: True values (used in simulation) and estimated values with their 95% confidence intervals. Equilibrium constants K_1, K_3, K_2, and K_8 might be numerically unstable to estimate: see confidence interval for K_2.

produce the sets of parameters which fit the experimental data perfectly. However these estimates do not have reliable confidence intervals. In other words, there are other sets of parameters that would fit the data equally well and, at the same time, have drastically different numerical values. Nevertheless, this allows us to use any of these sets to simulate the experimental data. In order to simulate the data we used the values from [51] and added some noise to the solution of the system. Then we estimated parameters according to the proposed algorithm outlined here. We used MATLAB (R2018a) for all numerical computations (see the code in Appendix B.3). The results of fitting are shown in Table 1. Examples of fitted model solutions are shown in Fig. 15.

Thus, from the simulated data we can conclude that the long–lived desensitized state is present (k_{13}^{+} is significantly greater than zero) and that the full—ligand state G_2RS_2 is more likely to unbind one of L–glutamate or D–serine molecule than the states GRS_2 or G_2RS, respectively (k_{12}^{-} is significantly greater than k_6^{-} and k_{11}^{-} is significantly greater than k_9^{-}). Thus, we correctly identified two independent sources of desensitization: presence of long–lived non–conductive state $G_2R'S_2$ and D–serine–dependent desensitization.

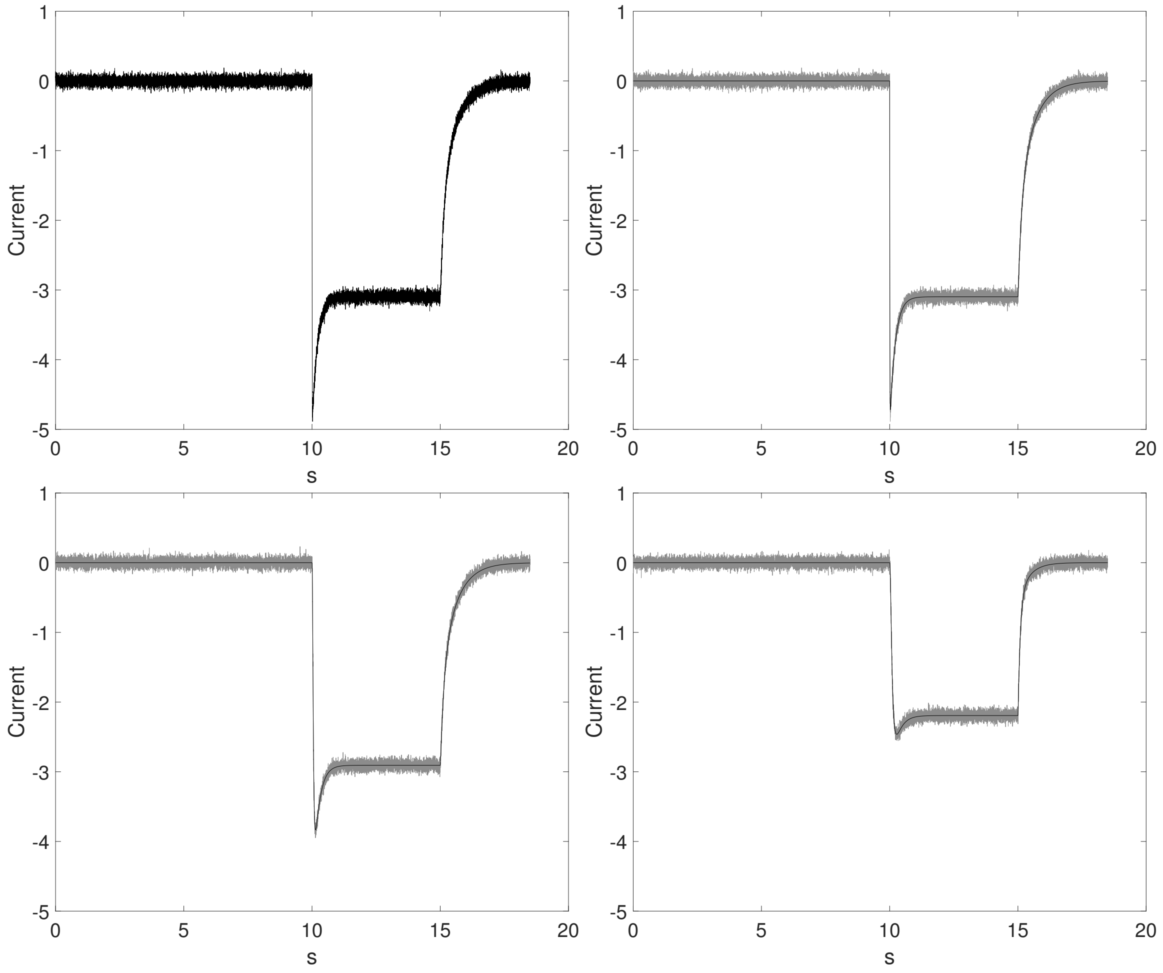

Figure 15: **Top left.** Example of a simulated current (black) in Type 1 experiment with both substrates being at a saturating levels ($10mM$ for both). The rest of the pictures correspond to the other examples of simulated current (gray) fitted with the corresponding models (black). **Top right.** Type 1 experiment with both substrates being at a saturating levels (10 mM for both). **Bottom left.** Type 1 experiment with saturating concentration of D–serine (10 mM) and low concentration of L–glutamate (10 μM). **Bottom right.** Type 2 experiment with saturating concentration of L–glutamate (10 mM) and low concentration of D–serine (10 μM).

5 Chaos Theory

Chaotic system is a deterministic dynamical system with high sensitivity to initial conditions and parameters, whose trajectories are bounded but are not periodic and do not tend to any steady–states. The slight change in initial conditions or parameter values leads to a long–term divergence of the "perturbed" solution from the solution of the original problem. This effect can be caused by such small variation in parameter values as those in rounding errors, different solver algorithms, or even the same algorithm but with minuscule changes in tolerance settings. This specific characteristic of chaotic systems lead to a problem of inability to estimate the system parameters via conventional means. The likelihood function cannot be constructed in a meaningful way: after a short initial period of time the small changes in parameters lead to arbitrary large changes in a solution. The standard Markov–Chain Monte–Carlo (MCMC) sampling can be applied but it results in a "Swiss cheese" style parameter posterior. A number of the recently published papers discuss this issue, see e.g., [53, 54, 55, 56].

A novel approach was introduced recently that allows one to construct a valid likelihood function based on a new concept of distance between the chaotic trajectories [57]. It was shown to produce robust and meaningful confidence regions for the parameters of various chaotic systems, which allows us to solve problems considered as "intractable and unsolved" in prior literature; see, e.g., [55]. Here we outline the approach and discuss its extension for systems with different time scales.

5.1 Correlation integral likelihood

The central idea of the approach is related to a generalized correlation integral concept. We remind the definition of a correlation integral: let $x_1, x_2, \ldots, x_N$ be the points of a trajectory $x \in \mathbb{R}^n$ at time points $t_1, t_2, \ldots, t_N$, respectively. For any $R > 0$ we can define a Correlation Integral sum

$$C(R, N) = \frac{1}{N^2} \sum_{i,j} \# \left(\|x_i - x_j\| < R \right), \tag{51}$$

where $\|\cdot\|$ denotes an Euclidean distance; $\#(\cdot)$ is equal to 1 if the condition within the parenthesis is satisfied and 0 otherwise. The correlation integral is then defined as

$$C(R) = \lim_{N\to\infty} C(R, N).$$

The approach discussed in [57] modifies the definition of integral sum (51) in order to use it as a measure of a distance between two different trajectories. Let $x = x(\theta, x_0)$ and $\widetilde{x} = \widetilde{x}(\widetilde{\theta}, \widetilde{x_0})$ be two solutions of the same model with different set of parameters θ, $\widetilde{\theta}$ and different initial conditions x_0, $\widetilde{x_0}$. Then for $R > 0$ the generalized correlation sum between the trajectories x and $\widetilde{x}$ is defined to be

$$C(R, N, \theta, x_0, \widetilde{\theta}, \widetilde{x_0}) = \frac{1}{N^2} \sum_{i,j} \# \left(\|x_i - \widetilde{x_j}\| < R \right), \tag{52}$$

where x_i and $\widetilde{x_j}$ are defined as before on the same time grid. By computing the generalized correlation sum for a fixed set of radii $R_1, R_2, \dots, R_K$, we obtain a K—dimensional vector for the trajectories. Since each component of the vector is an average, by the Central Limit Theorem, the vector is expected to be Gaussian. Then one constructs a distribution of a "natural" variability in the solutions due to their chaotic nature, i.e., variability in $C(R_i, N, \theta, x_0, \theta, \widetilde{x_0})$ $(i = 1, \dots, K)$ for various initial conditions. Then one can compare this variability with variability due to the parameter variation, i.e., $C(R_i, N, \theta, x_0, \widetilde{\theta}, \widetilde{x_0})$ $(i = 1, \dots, K)$, using MCMC approach. The idea is that variation in initial conditions of the system changes the solution of the system but does not change the chaotic attractor, while variation in parameters does change the properties of the attractor. Here we do not go into more details of the methodology because this is not needed for further discussion.

5.2 Parameter estimation in chaotic systems with different time scales using the correlation integral approach

The described method constructs a likelihood based on the generalized correlation integral and allows one to compare trajectories of chaotic systems. However, because the whole solution gets mapped into a vector, some information is obviously lost, e.g., the time dependent dynamics of the system. Let us consider a case when variables of the system act on significantly different time scales. In particular, we assume that the fast dynamics corresponds to a stable transition of the solution trajectories starting "outside" of the "slow" chaotic attractor, approaches it, and stays on it. This situation rises two potential problems for the described methodology. First, in order to construct the appropriate distribution of mapped vectors that corresponds to "natural" variation, all initial conditions must start on the attractor. Usually, the authors simply "cut—off" some initial period of each solution to resolve this problem. While it might work for simple systems, generally speaking the length of such "initial" period is unknown. Second, if we are interested in parameters that regulate the transition of the solution to the attractor, they may be impossible to estimate once the solution reaches the attractor. In both cases it is important to find a methodology that can answer how close a given trajectory point is to the attractor. Here we discuss such procedure (based on the one from Appendix A.4), which allows one to check if a particular point of a solution has a chaotic attractor in its vicinity.

For the sake of clarity we discuss a special case of a system with a single fast variable that adjusts to the behavior of one chaotic variable. However, this approach can be easily extended to a more general case. Let us consider a chaotic system with slow variables

$$\frac{dx}{dt} = f(x),$$

where $x = (x_1, x_2, \ldots, x_n)$ and $f = (f_1, f_2, \ldots, f_n)$, where all components of f are differentiable. We introduce a fast variable w which adjusts to the variable x_1 in the following way

$$\begin{aligned} \frac{dx}{dt} &= f(x), \\ \frac{dw}{dt} &= -K(w - x_1), \end{aligned}$$

with a positive constant $K \gg 1$. Let us perform the change of variables $v = w - x_1$. Then the system becomes

$$\begin{aligned} \frac{dx}{dt} &= f(x), \\ \frac{dv}{dt} &= -Kv - f_1(x). \end{aligned}$$

If we are interested in whether a point (x^0, w^0) is at most at distance R from the attractor or not, we find

$$h^0 = \begin{pmatrix} f(x^0) \\ \\ -Kv^0 - f_1(x^0) \end{pmatrix}, \qquad J^0 = \begin{pmatrix} J_x^0 & 0 \\ \\ 0 & -K \end{pmatrix},$$

where $v^0 = w^0 - x_1^0$ and J_x^0 is a Jacobian of $f(x)$ at x^0. There is a transformation matrix T_x such that $S_x^0 = T_x^{-1} J_x T^x$ is a block diagonal matrix. Thus, matrix

$$T = \begin{pmatrix} T_x & 0 \\ 0 & 1 \end{pmatrix}$$

is the transformation matrix such that $T^{-1} J^0 T$ is block diagonal:

$$T^{-1} J^0 T = \begin{pmatrix} T_x^{-1} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} J_x^0 & 0 \\ 0 & -K \end{pmatrix} \begin{pmatrix} T_x & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} S_x^0 & 0 \\ 0 & -K \end{pmatrix}.$$

Following the procedure from Appendix A.4, we can now choose $\nu = K$. Since $K \gg 1$, all eigenvalues of S_x^0 are greater than $-K$. Thus, $S_{11}^0 = S_x^0$ and $S_{22}^0 = (-K)$. It is obvious that the inequality

$$\frac{\|S_{11}^0\|}{K} < 1$$

is always true. Thus, the only condition left to be checked is

$$\frac{|\widehat{h}_2^0|}{K} < R,$$

where $\widehat{h}_2^0$ is the bottommost entry of vector $T^{-1}h^0$, i.e., $\widehat{h}_2^0 = -Kv^0 - f(x^0) = -K(w^0 - x_1^0) - f(x^0)$. The condition becomes

$$\frac{1}{K}\left|K(w^0 - x_1^0) + f(x^0)\right| < R. \tag{53}$$

As an example, let us consider the following extended Lorenz63 chaotic system:

$$\begin{aligned} \frac{dx}{dt} &= \beta(y - x), \\ \frac{dy}{dt} &= x(\gamma - z), \\ \frac{dz}{dt} &= xy - \alpha z, \\ \frac{dw}{dt} &= -K(w - x). \end{aligned} \tag{54}$$

For $v = w - x$, the last equation of the above system can be rewritten as:

$$\frac{dv}{dt} = -Kv - \beta(y - x), \tag{55}$$

where $\alpha = 8/3$, $\beta = 10$, $\gamma = 28$, and $K = 1000$. We take the following initial conditions $x(0) = 2.51$, $y(0) = 2.51$, $z(0) = 19.92$, and $v(0) = -20$. In that case (x, y, z) lie on the chaotic attractor and v rapidly approaches its stable manifold $v = 0$ (as w rapidly adjusts to x). We take the following time points $t_1 = 1.0005$, $t_2 = 1.0015$, $t_3 = 1.0025$, and $t_4 = 1.0035$ and check whether the corresponding points v_1, v_2, v_3, and v_4 are near the chaotic attractor within radius $R = 2$ or not; see Figure 16.

The methodology described here can be automatically implemented to complement the generalized correlation integral approach of estimation of chaotic system's parameters. This will allow for automating a crucial step of the estimation.

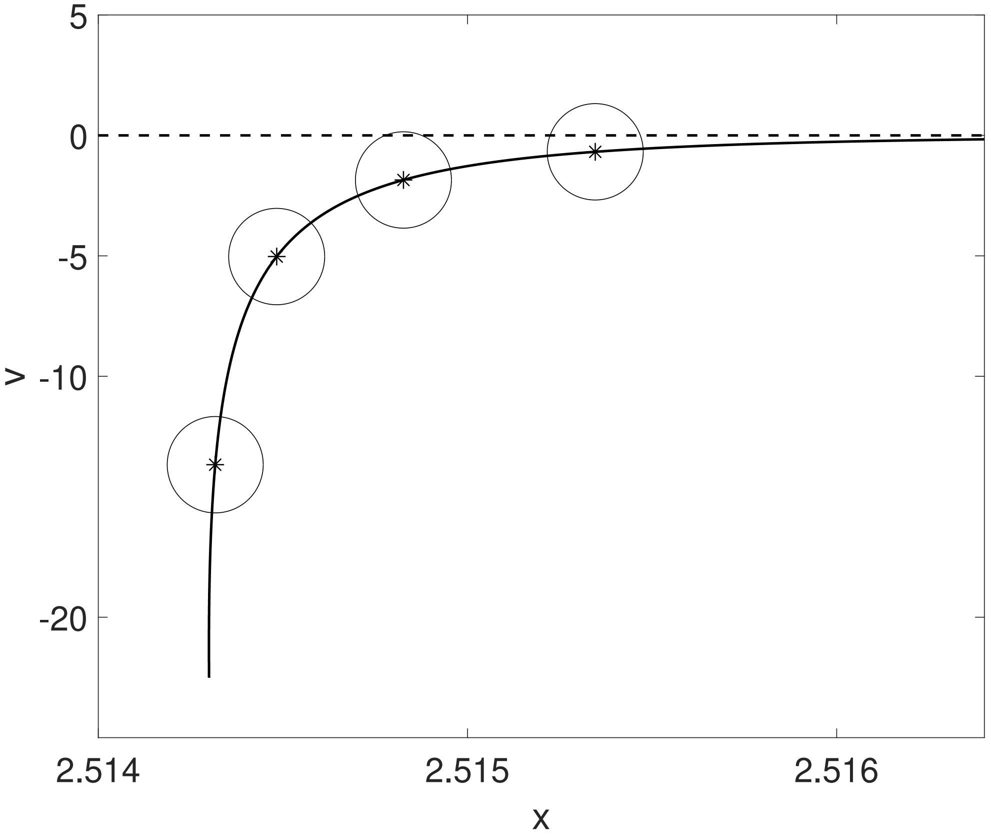

Figure 16: The solid line represents the solution of system (54), (55) projected on an (x, v) plane and the dashed line is the corresponding stable manifold ($v = 0$). The solution is already on the chaotic attractor for x, y, z variables. The selected points on the trajectory at time points t_1, t_2, t_3, t_4 are marked with the stars. Each selected point has a circle drawn around it (radius $R = 2$). Note that in reality the projection of each ball of radius 2 onto (x, v) plane would look like and ellipse on the graph (see scale of x—axis). However, since we are interested in the proximity of a solution approaching plane $v = 0$ along v—axis, we decided to draw circles instead to make the plot more visual. The corresponding values of left—hand side of inequality condition (53) for t_1, t_2, t_3, t_4 are 13.67, 5.03, 1.85, and 0.68, respectively. Thus, the condition (53) is true only for point $v_3 = v(t_3)$ and $v_4 = v(t_4)$ as expected.

6 Conclusion

In this dissertation, we constructed a chemical kinetics model that allowed us to obtain statistically reliable estimates of the glutamate transporters' turnover rates (in the current literature the reliability regions for the parameter values are rarely estimated). We used estimated values to create a 3–dimensional model of a single synapse and simulate synaptic transmission. We concluded that, based on the simulation, transporters do not play a significant role in clearing the synaptic cleft from the neurotransmitter glutamate. The effects observed during whole–cell recording experiments or field Excitatory Postsynaptic Potential recordings are due to some other effects, e.g., an activation of extrasynaptic receptors and spillover phenomenon.

We also constructed a series of models alongside with series of experiments that, if used on conjunction, allow for a statistically reliable estimates of some parameters of N–methyl–D–aspartate receptors. We do not have experimental data to fully answer the question about the nature of NMDARs' desensitization but we demonstrated the effectiveness of the proposed algorithm on a simulated data.

Finally, we discussed a novel approach of estimating parameters of chaotic systems and its possible improvement.

References

[1] Lin C. L., Bristol L. A., Jin L., Dykes-Hoberg M., Crawford T., Clawson L., Rothstein J. D., *Aberrant RNA processing in a neurodegenerative disease: the cause for absent EAAT2, a glutamate transporter, in amyotrophic lateral sclerosis*, Neuron, 1998 Mar; 20:589—602.

[2] Scott H. A., Gebhardt F. M., Mitrovic A. D., Vandenberg R. J., Dodd P. R., *Glutamate transporter variants reduce glutamate uptake in Alzheimer's disease*, Neurobiol Aging, 2011 Mar; 32:553 e1—11.

[3] Zhou Y. and Danbolt N. C., *Glutamate as a neurotransmitter in the healthy brain*, J Neural Transm, 2014 Aug; 121:799—817.

[4] Tzingounis A. V., Wadiche J. I., *Glutamate transporters: confining runaway excitation by shaping synaptic transmission*, Nat Rev Neurosci., 2007 Dec; 8(12):935—47.

[5] Ventura R. and Harris K. M., *Three-Dimensional Relationships between Hippocampal Synapses and Astrocytes*, The Journal of Neuroscience, 1999 Aug; 19(16):6897—6906.

[6] Danbolt N. C., Furness D. N., Zhou Y., *Neuronal vs glial glutamate uptake: Resolving the conundrum*, Neurochemistry International, 2016 Sep; 98:29—45.

[7] Holmseth S., Dehnes Y., Huang Y. H., Follin–Arbelet V. V., Grutle N. J., Mylonakou M. N., Plachez C., Zhou Y., Furness D. N., Bergles D. E., Lehre K. P., and Danbolt N. C., *The Density of EAAC1 (EAAT3) Glutamate Transporters Expressed by Neurons in the Mammalian CNS*, The Journal of Neuroscience, 2012 Apr; 32(17):6000—6013.

[8] Wadiche J. I., Arriza J. L., Amara S. G., and Kavanaugh M. P., *Kinetics of a Human Glutamate Transporter*, Neuron, 1995 May; 14:1019—1027.

[9] Wadiche J. I. and Kavanaugh M. P., *Macroscopic and Microscopic Properties of a Cloned Glutamate Transporter/Chloride Channel*, the Journal of Neuroscience, 1998 Oct; 18(19):7650—7661.

[10] Grewer C., Watzke N., Wiessner M., Rauen T., *Glutamate translocation of the neuronal glutamate transporter EAAC1 occurs within milliseconds*, PNAS, 2000 Aug; 97(17):9706—9711.

[11] Bergles D. E., Jahr C. E., *Glial contribution to glutamate uptake at Schaffer collateral-commissural synapses in the hippocampus*, Journal of Neurophysiology, 1998 Oct; 18(19):7709—16.

[12] Zerangue N., Kavanaugh M. P., *Flux coupling in a neuronal glutamate transporter*, Nature, 1996 Oct; 383(6601):634—7.

[13] Levy L. M., Warr O., Attwell D., *Stoichiometry of the glial glutamate transporter GLT-1 expressed inducibly in a Chinese hamster ovary cell line selected for low endogenous Na+-dependent glutamate uptake*, J Neurosci., 1998 Dec; 18(23):9620—8.

[14] Larsson H. P., Tzingounis A. V., Koch H. P., and Kavanaugh M. P., *Fluorometric measurements of conformational changes in glutamate transporters*, PNAS, 2004 Mar; 101, 3951—3956.

[15] Bergles D. E., Tzingounis A. V., and Jahr C. E., *Comparison of Coupled and Uncoupled Currents during Glutamate Uptake by GLT-1 Transporters*, The Journal of Neuroscience, 2002 Dec; 22(23):10153—10162.

[16] Wadiche J. I., Amara S. G., Kavanaugh M. P., *Ion fluxes associated with excitatory amino acid transport*, Neuron, 1995 Sep; 15:721—728.

[17] Otis T. S. and Kavanaugh M. P., *Isolation of Current Components and Partial Reaction Cycles in the Glial Glutamate Transporter EAAT2*, The Journal of Neuroscience, 2000 Apr; 20(8):2749—2757.

[18] Borre L., Kavanaugh M. P., and Kanner B. I., *Dynamic Equilibrium between Coupled and Uncoupled Modes of a Neuronal Glutamate Transporter*, J Biol Chem., 2002 Apr; 277(16):13501—7.

[19] Haario H., Laine M., Mira A., Saksman E., *DRAM: Efficient adaptive MCMC*, Statistics and Computing, 2006 Dec; 16(4):339—354.

[20] MCMC toolbox for Matlab by Marko Laine `http://helios.fmi.fi/~lainema/mcmc/`.

[21] Clements J. D., Lester R. A., Tong G., Jahr C. E., Westbrook G. L., *The time course of glutamate in the synaptic cleft*, Science, 1992 Nov; 258(5087):1498—501.

[22] Tong G., Jahr C. E., *Multivesicular release from excitatory synapses of cultured hippocampal neurons*, Neuron, 1994 Jan; 12(1):51—9.

[23] Otis T. S., Wu Y. C., Trussell L. O., *Delayed clearance of transmitter and the role of glutamate transporters at synapses with multiple release sites*, J Neurosci., 1996 Mar; 16(5):1634—44.

[24] Asztely F., Erdemli G., Kullmann D. M., *Extrasynaptic glutamate spillover in the hippocampus: dependence on temperature and the role of active glutamate uptake*, Neuron, 1997 Feb; 18:281—293.

[25] Lozovaya N. A., Kopanitsa M. V., Boychuk Y. A., Krishtal O. A., *Enhancement of glutamate release uncovers spillover-mediated transmission by N-methyl-D- aspartate receptors in the rat hippocampus*, Neuroscience, 1999 Oct; 91:1321—1330.

[26] Arnth–Jensen N., Jabaudon D., Scanziani M., *Cooperation between independent hippocampal synapses is controlled by glutamate uptake*, Nat. Neurosci, 2002 Apr; 5:325—331.

[27] Sun W., Hoffman K. M., Holley D. C., Kavanaugh M. P., *Specificity and Actions of an Arylaspartate Inhibitor of Glutamate Transport at the Schaffer Collateral-CA1 Pyramidal Cell Synapse*, PLoS One, 2011 Aug; 6(8):e23765.

[28] Le Meur K., Galante M., Angulo M. C., Audinat E., *Tonic activation of NMDA receptors by ambient glutamate of non-synaptic origin in the rat hippocampus*, J Physiol., 2007 Apr; 580(Pt. 2):373—83.

[29] Ribeiro A. C. F., Rodrigo M. M., Barros M. C. F., Verissimo L. M. P., Romero C., Valente A. J. M., Esteso M. A., *Mutual diffusion coefficients of L-glutamic acid and monosodium L-glutamate in aqueous solutions at T = 298.15 K*, J. Chem. Thermodynamics, 2014 July; 74:133—137.

[30] Savtchenko L. P. and Rusakov D. A., *The optimal height of the synaptic cleft*, PNAS, 2007 Feb; 104(6):1823—1828.

[31] Zheng K., Jensen T. P., Savtchenko L. P., Levitt J. A., Suhling K., Rusakov D. A., *Nanoscale diffusion in the synaptic cleft and beyond measured with time-resolved fluorescence anisotropy imaging*, Sci Rep., 2017 Feb; 7:42022.

[32] Arriza J. L., Fairman W. A., Wadiche J. I., Murdoch G. H., Kavanaugh M. P., and Amara S. G., *Functional comparisons of three glutamate transporter subtypes cloned from human motor cortex*, Journal of Neuroscience, 1994 Sep; 14(9):5559—5569.

[33] Jabaudon D., Shimamoto K., Yasuda-Kamatani Y., Scanziani M., Gähwiler B. H., and Gerber U., *Inhibition of uptake unmasks rapid extracellular turnover of glutamate of nonvesicular origin*, Proc Natl Acad Sci USA, 1999 Jul; 96(15): 8733—8738.

[34] Cavelier P., Attwell D., *Tonic release of glutamate by a DIDS-sensitive mechanism in rat hippocampal slices*, J Physiol., 2005 Apr; 564(Pt 2):397—410.

[35] Herman M. A., Jahr C. E., *Extracellular glutamate concentration in hippocampal slice*, J Neurosci, 2007 Sep; 27:9736—9741.

[36] Sun W., Shchepakin D., Kalachev L. V., Kavanaugh M. P., *Glutamate transporter control of ambient glutamate levels*, Neurochemistry International, 2014 Jul; 73:146—151.

[37] Rusakov D. A. and Kullmann D. M., *Extrasynaptic Glutamate Diffusion in the Hippocampus: Ultrastructural Constraints, Uptake, and Receptor Activation*, The Journal of Neuroscience, 1998 May; 18(9):3158—3170.

[38] Hessler N. A., Shirke A. M., and Malinow R., *The probability of transmitter release at a mammalian central synapse*, Nature, 1993 Dec; 366:569—572.

[39] Riveros N., Fiedler J., Lagos N., Muñoz C., Orrego F., *Glutamate in rat brain cortex synaptic vesicles: influence of the vesicle isolation procedure*, Brain Research, 1986 Oct; 386(1—2):405—408.

[40] Mothet J., Parent A. T., Wolosker H., Brady R. O. Jr., Linden D. J., Ferris C. D., Rogawski M. A., and Snyder S. H., *D–serine is an endogeneous ligand for the glycine site of the N—methyl—D—aspartate receptor*, Proc Natl Acad Sci USA, 2000 Apr; 97(9):4926—4931.

[41] Mayer M. L., Vyklicky L. Jr., and Clements J., *Regulation of NMDA receptor desensitization in mouse hippocampal neurons by glycine*, Nature, 1989 Mar; .338(6214):425—-427.

[42] Benveniste M., Clements J., Vyklicky L. Jr., and Mayer M. L. *A kinetic analysis of the modulation of N—methyl—D—aspartic acid receptorsby glycine in mouse cultured hippocampal neurones*, J. Physiol, 1990 Sep; 428:333—-357.

[43] Vyklický L. Jr., Benveniste M., Mayer M. L., *Modulation of N-methyl-D-aspartic acid receptor desensitization by glycine in mouse cultured hippocampal neurones*, J Physiol, 1990 Sep; 428:313—331.

[44] Sather W., Johnson J. W., Henderson G., Ascher. P., *Glycine—insensitive desensitization of NMDA responses in cultured mouse embryonic neurons*, Neuron, 1990 May; 4:725—731.

[45] Legendre P., Rosenmund C., and Westbrook. G. L., *Inactivation of NMDA channels in cultured hippocampal neurons by intracellular calcium*, J. Neurosci., 1993 Feb; 13(2):674—684.

[46] Tong G., Jahr C. E., *Regulation of glycine—insensitive desensitization of the NMDA receptor in outside—out patches*, J. Neurophysiol., 1994 Aug; 72(2):754—761.

[47] Medina I., Filippova N., Charton G., Rougeole S., Ben-Ari Y., Khrestchatisky M., and Bregestovski P., *Calcium—dependent inactivation of heteromeric NMDA receptor—channels expressed in human embryonic kidney cells*, J. Physiol., 1995, Feb; 482:567—573.

[48] Krupp J. J., Vissel B., Heinemann S. F., and Westbrook G. L., *Calcium—dependent inactivation of recombinant N–methyl–D–aspartate receptors is NR2 subunit specific*, Mol. Pharmacol., 1996 Dec; 50:1680—1688.

[49] Krupp J. J., Vissel B., Heinemann S. F., and Westbrook G. L., *N—terminal domains in the NR2 subunit control desensitization of NMDA receptors*, Neuron, 1998 Feb; 20:317—327.

[50] Krupp J. J., Vissel B., Thomas C. G., Heinemann S. F., and Westbrook G. L., *Interactions of calmodulin and alpha—actinin with the NR1 subunit modulate Ca2+—dependent inactivation of NMDA receptors*, J. Neurosci., 1999 Feb; 19:1165—1178.

[51] Nahum-Levy R., Lipinski D., Shavit S., Benveniste M., *Desensitization of NMDA receptor channels is modulated by glutamate agonists*, Biophys. J., 2001 May; 80(5):2152—2166.

[52] Cummings K. A., Popescu G. K., *Glycine—dependent activation of NMDA receptors*, J Gen Physiol 2015 Jun; 145(6):513—527.

[53] Peng B., Liu B., Zhang F.–Y., Wang L., *Differential evolution algorithm—based parameter estimation for chaotic systems*, Chaos, Solitons & Fractals, 2009 March; 39(5):2110—2118.

[54] Ho W.–H., Chou J.–H., Guo C.–Y., *Parameter identification of chaotic systems using improved differential evolution algorithm*, Nonlinear Dynamics, 2010 July; 61(1—2):29—-41.

[55] Rougier J., *"Intractable and unsolved": some thoughts on statistical data assimilation with uncertain static parameters*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2013 May; 371(1991).

[56] Li X., Yin M., *Parameter estimation for chaotic systems by hybrid differential evolution algorithm and artificial bee colony algorithm*, Nonlinear Dynamics, 2014 July; 77(1—2):61—71.

[57] Haario H., Kalachev L., Hakkarainen J., *Generalized correlation integral vectors: A distance concept for chaotic dynamical systems*, Chaos, 2015 Jun; 25(6):063102.

[58] A. N. Tikhonov, A. B. Vasil'eva, and A. G. Sveshnikov. Differential Equations, Springer Series in Soviet Mathematics, Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 1985.

[59] Vasil'eva A. B., Butuzov V. F., and Kalachev L. V., *The Boundary Function Method for Singular Perturbation Problems*, SIAM Studies in Applied Mathematics, Philadelphia, PA, 1995.

[60] Kline M., *Mathematical thought from ancient to modern times*, Vol. 2, Oxford University Press, 1972.

[61] Handrock-Meyer S., Kalachev L. V., Schneider K.R., *A Method to Determine the Dimension of Long-Time Dynamics in Multi-Scale Systems*, Journal of Mathematical Chemistry, 2001 Aug; 30(2):133—160.

Appendix

A Useful results and methods

A.1 Asymptotic methods

Suppose, we have a system

$$\begin{aligned} \varepsilon\frac{d\mathbf{z}}{dt} &= \mathbf{F}(t,\mathbf{z},\mathbf{y}). \\ \frac{d\mathbf{y}}{dt} &= \mathbf{G}(t,\mathbf{z},\mathbf{y}), \end{aligned} \tag{56}$$
$$\mathbf{z}(0,\varepsilon) = \mathbf{z}^0, \quad \mathbf{y}(0,\varepsilon) = \mathbf{y}^0,$$

where $0 < \varepsilon \ll 1$ is a small parameter and time t varies from 0 to T. Here $\mathbf{z}$ and $\mathbf{y}$ are vectors, $\mathbf{F}$ and $\mathbf{G}$ are vector functions. Moreover, the functions $\mathbf{F}(t,\mathbf{z},\mathbf{y})$ and $\mathbf{G}(t,\mathbf{z},\mathbf{y})$ are continuous together with their derivatives in some domain

$$\Omega = \{\|\mathbf{z}\| \leq a, \|\mathbf{y}\| \leq a, 0 \leq t \leq T\}.$$

Along with the system (56) we will consider the system

$$\begin{aligned} 0 &= \mathbf{F}(t,\bar{\mathbf{z}},\bar{\mathbf{y}}). \\ \frac{d\bar{\mathbf{y}}}{dt} &= \mathbf{G}(t,\bar{\mathbf{z}},\bar{\mathbf{y}}), \end{aligned} \tag{57}$$
$$\bar{\mathbf{y}}(0) = \mathbf{y}^0,$$

and call it a reduced system.

Now, let us introduce the following conditions:

1. Let $\mathbf{F}(t,\bar{\mathbf{z}},\bar{\mathbf{y}}) = 0$ have an isolated root $\bar{\mathbf{z}}(t) = \boldsymbol{\phi}(t,\bar{\mathbf{y}}(t))$, $(t,\bar{\mathbf{y}}) \in D = \{\|\bar{\mathbf{y}}\| \leq a, 0 \leq t \leq T\}$, and suppose that the corresponding reduced problem has a unique solution in the interval $0 \leq t \leq T$.

2. The steady–state $\widetilde{\mathbf{z}} = \boldsymbol{\phi}(t,\bar{\mathbf{y}})$ of the system

$$\frac{d\widetilde{\mathbf{z}}}{d\tau} = \mathbf{F}(t,\widetilde{\mathbf{z}},\bar{\mathbf{y}}), \quad \tau \geq 0, \tag{58}$$

   with parameters t and $\widetilde{\mathbf{y}}$, is asymptotically stable in the sense of Lyapunov, uniformly in $(t,\bar{\mathbf{y}}) \in D$ as $\tau \to \infty$.

3. Let the solution $\widetilde{\mathbf{z}}(\tau)$ of the problem (58) for $\widetilde{\mathbf{z}}$ with $\bar{\mathbf{y}} = \mathbf{y}^0$ and $t = 0$ exist for $\tau \geq 0$ and tend to the stationary point $\boldsymbol{\phi}(0,\mathbf{y}^0)$ as $\tau \to \infty$.

Theorem 1 (Tikhonov's theorem [58, 59]) *Under conditions 1 — 3 and for sufficient small ε, the original perturbed problem (56) has a unique solution $\boldsymbol{z}(t,\varepsilon)$, $\boldsymbol{y}(t,\varepsilon)$ such that the following limiting equalities hold:*

$$\lim_{\varepsilon\to 0} \boldsymbol{y}(t,\varepsilon) = \bar{\boldsymbol{y}}(t), \qquad \textit{for } 0 < t \le T,$$

$$\lim_{\varepsilon\to 0} \boldsymbol{z}(t,\varepsilon) = \bar{\boldsymbol{z}}(t), \qquad \textit{for } 0 \le t \le T,$$

Now, let us consider one more condition.

4. The functions $\mathbf{F}(t,\mathbf{z},\mathbf{y})$ and $\mathbf{G}(t,\mathbf{z},\mathbf{y})$ are infinitely differentiable in the domain Ω.

And let us define the asymptotic series of some vector or scalar function $\mathbf{x}(t,\varepsilon)$ as

$$\mathbf{x}(t,\varepsilon) = \bar{\mathbf{x}}(t,\varepsilon) + \mathbf{\Pi}\mathbf{x}(\tau,\varepsilon),$$

where $\tau = t/\varepsilon$ and

$$\bar{\mathbf{x}}(t,\varepsilon) = \bar{\mathbf{x}}_0(t) + \varepsilon\bar{\mathbf{x}}_1(t) + \cdots + \varepsilon^k\bar{\mathbf{x}}_k(t) + \cdots$$

is called the regular part of the expansion and each individual function is called a regular function,

$$\mathbf{\Pi}\mathbf{x}(t,\varepsilon) = \mathbf{\Pi}_0\mathbf{x}(\tau) + \varepsilon\mathbf{\Pi}_1\mathbf{x}(\tau) + \cdots + \varepsilon^k\mathbf{\Pi}_k\mathbf{x}(\tau) + \cdots$$

is called the boundary layer part and each individual function is called a boundary function.

Because the boundary layer part corresponds to the behavior of the function $\mathbf{x}(t,\varepsilon)$ near $t = 0$, the following condition holds

$$\lim_{\tau\to\infty} \mathbf{\Pi}_k\mathbf{x}(\tau) = 0.$$

Finally, let us define the partial sum of the asymptotic series as

$$\mathbf{X}_n(t,\varepsilon) = \sum_{k=0}^{n} \varepsilon^k[\bar{\mathbf{x}}_k(t) + \mathbf{\Pi}_k\mathbf{x}(\tau)].$$

We introduce the following theorem.

Theorem 2 (Vasil'eva's theorem [58, 59]) *Under conditions 1 — 4, for the asymptotic series of the solution $\boldsymbol{x}(t,\varepsilon) = \{\boldsymbol{z}(t,\varepsilon), \boldsymbol{y}(t,\varepsilon)\}$ of the original perturbed problem in the interval $0 \le t \le T$, the following estimate holds:*

$$\max_{0\le t\le T} \|\boldsymbol{x}(t,\varepsilon) - \boldsymbol{X}_n(t,\varepsilon)\| = O(\varepsilon^{n+1}).$$

A.2 Modeling chemical kinetics on the boundary of a free diffusion region

Suppose we have a free diffusion process defined in some convex region $\Theta \subset \mathbb{R}^3$:

$$\frac{\partial y}{\partial t} = D\Delta y, \qquad t \geq 0, \qquad x \in \Theta,$$

where $y(t,x)$ is the concentration of the substance of interest; D is a diffusion coefficient. Suppose $S \subset \partial\Theta$ is some connected portion of a boundary of the region. We also assume that a number of chemical reactions occur in a small volume in a proximity of S, i.e. in a volume P, defined by a Cartesian product $S \times \omega \mathbf{n}$, where $\mathbf{n}$ is a normal vector to S and ω is a small number that characterizes the width of the volume P. We further assume that these chemical reactions can be described by a chemical kinetics system of ordinary differential equations (ODEs) and include the substance of interest as one of the compounds:

$$\begin{aligned} \frac{\partial \widetilde{y}}{\partial t} &= f(\widetilde{y}, z_1, \ldots, z_n), \\ \frac{dz_1}{dt} &= g_1(\widetilde{y}, z_1, \ldots, z_n), \\ &\vdots \\ \frac{dz_n}{dt} &= g_n(\widetilde{y}, z_1, \ldots, z_n), \end{aligned} \tag{59}$$

where $\widetilde{y}$ is a concentration of a substance of interest; $z_1, \ldots, z_n$ are other compounds that interact with the substance of interest; $f, g_1, \ldots g_n$ are some functions that describe the interaction of substances according to the stoichiometric relationships and the law of mass action.

If ω is small enough, then we can assume that as soon as the substance of interest enters volume P from volume Θ via a free diffusion, it is well—mixed in P and takes part in the reactions described by (59). The substance flux from Θ to P is defined by Fick's first law of diffusion. The total concentration of a substance entering and leaving through the surface S per unit of time is

$$-D\frac{1}{\omega}\frac{\partial y}{\partial \mathbf{n}}.$$

At the same time, the amount of produced and consumed substance due to the chemical reactions is given by (59). Since these two formulas describe the same substance, their equality must hold at the surface S:

$$-D\frac{\partial y}{\partial \mathbf{n}} = \omega f(y, z_1, \ldots, z_n),$$

which produces Neumann type boundary conditions for the diffusion equation, where $z_1, \ldots z_n$ are the solutions of (59) with $\widetilde{y} = y$.

A.3 Finite difference approximation

Let us consider a k times differentiable (on some interval (a, b)) function $f(x)$. We can apply Taylor's theorem [60] to it at an arbitrary point $x_0 \in (a, b)$:

$$f(x_0 + h) = f(x_0) + \frac{f'(x_0)}{1!}h + \frac{f''(x_0)}{2!}h^2 + \cdots + \frac{f^{(k)}(x_0)}{k!}h^k + O(h^{k+1}), \qquad (60)$$

and

$$f(x_0 - h) = f(x_0) - \frac{f'(x_0)}{1!}h + \frac{f''(x_0)}{2!}h^2 - \cdots + (-1)^k\frac{f^{(k)}(x_0)}{k!}h^k + O(h^{k+1}), \qquad (61)$$

where $h > 0$ is a step size of discretization. Therefore, subtracting (61) from (60) we get:

$$f(x_0 + h) - f(x_0 - h) = f'(x_0)2h + O(h^3),$$

and so,

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} + O(h^2). \qquad (62)$$

Taking the sum of equations (60) and (61), we obtain

$$f(x_0 + h) + f(x_0 - h) = 2f(x_0) + f''(x_0)h^2 + O(h^4),$$

and hence,

$$f''(x_0) = \frac{f(x_0 + h) - 2f(x_0) + f(x_0 - h)}{h^2} + O(h^2). \qquad (63)$$

Thus, if a function f is differentiable in some interval, we can approximate its first and second derivatives at any point inside that interval with the first terms of the right hand sides of the equations (62) and (63), respectively.

A.4 A method to check the presence of an invariant manifold in the proximity of a point in the phase space

Suppose we have a system

$$\frac{dz}{dt} = h(z,t), \qquad (z,t) \in \mathbb{R}^n \times \mathbb{R}, \tag{64}$$

where h is twice continuously differentiable with respect to z and t. Let $R > 0$ be a sufficiently small constant. In order to answer whether a point (z_0, t_0) has an invariant manifold of system (64) within the ball of radius R we need to follow the following procedure [61]:

1. Find value of function h and its Jacobian at the point of interest: $h^0 = h(z_0, t_0)$ and $J^0 = h_z(z_0, t_0)$. Note that $\widetilde{h}(z, t, z_0, t_0) = h(z,t) - h^0 - J^0(z - z_0) = O(|z - z_0|^3 + |t - t_0|)$.

2. Find a transformation matrix T that block diagonalizes matrix J^0, i.e., $S^0 = T^{-1}J^0T$ is a block diagonal matrix. Moreover, among all possible matrices T choose such that $S^0 = \text{diag}\left(S^0_{11}, S^0_{22}\right)$, where all eigenvalues of S^0_{11} are greater than $-\nu$ and all eigenvalues of S^0_{22} are less than or equal to $-\nu$, where $\nu > 0$ is some constant.

3. Apply the following substitution of variables: $z = z_0 + Tu$.

4. Rewrite the system (64) in terms of the variable u:

   $$\frac{du}{dt} = T^{-1}h^0 + S^0u + T^{-1}\widetilde{h}(z_0 + Tu, t, z_0, t_0).$$

   Taking into account the block diagonal structure of $S^0 = diag\left(S^0_{11}, S^0_{22}\right)$, the system can be further rewritten in the following form:

   $$\frac{du_1}{dt} = \widehat{h}^0_1 + S^0_{11}u_1 + \bar{h}_1(u, t, z_0, t_0),$$
   $$\frac{du_2}{dt} = \widehat{h}^0_2 + S^0_{22}u_2 + \bar{h}_2(u, t, z_0, t_0).$$

5. Verify the following inequalities:

   $$\frac{\|S^0_{11}\|}{\nu} < 1, \qquad \frac{|\widehat{h}^0_2|}{\nu} < R,$$

   where $\|\cdot\|$ is the matrix norm induced by the euclidean vector norm, i.e., $\|A\| = \sqrt{\rho(A^TA)}$, where ρ denotes the spectral radius. If the inequalities are satisfied, then there is an invariant manifold of system (64) within the ball of radius R and center at (z_0, t_0).

B MATLAB codes

B.1 Glutamate transporter chemical kinetics model

EAAT/run.m

```
addpath ./mcmcstat

run_mcmc = true;
nonlinfit = false;
add_peaks = false;
add_overshoots = false;
transporter = 'e1';
nsimu = 1000;

load(sprintf('%s.mat', transporter))
time = eval(sprintf('%s.time', transporter));
current = eval(sprintf('%s.traces', transporter));
beta0 = eval(sprintf('%s.beta0', transporter));
indeces = eval(sprintf('%s.indeces', transporter));
try
    overshoots = eval(sprintf('%s.overshoot_ind', transporter));
end
pulses = arrayfun(@(i) time{i}(indeces{i}), 1:length(indeces), 'Uni', 0);
for i = 1:length(current)
    m = mean(current{i}(indeces{i}(1):indeces{i}(2)));
    current{i} = current{i} - m;
end
% add weight to peaks
if add_peaks || add_overshoots
    switch transporter
        case 'e2'
            add_spike_scale = 2;
            add_overshoot_scale = 1;
        case 'e3'
            add_spike_scale = 4;
        otherwise
            add_spike_scale = 2;
    end
    for i = 1:length(current)
        if add_overshoots
            % overshoot
            for j = 3:2:length(indeces{i})
                start = overshoots{i}(j-2);
                finish = overshoots{i}(j-1);
                ovs = current{i}(start:finish);
                l = length(ovs);
                n = round((indeces{i}(j+1) - 1 - indeces{i}(j)) /...
                                                 add_overshoot_scale / l);
                timeshoot = repmat(time{i}(start:finish)', n, 1);
                time{i} = [time{i}(1:start-1); timeshoot(:);...
                                                    time{i}(finish+1:end)];
                ovs = repmat(ovs', n, 1);
                current{i} = [current{i}(1:start-1); ovs(:);...
                                                current{i}(finish+1:end)];
                indeces{i}(j+1:end) = indeces{i}(j+1:end) + (n - 1) * 1;
                overshoots{i}(j:end) = overshoots{i}(j:end) + (n - 1) * 1;
            end
        end

        if add_peaks
            % peaks
            for j = 2:2:length(indeces{i})-1
                start = indeces{i}(j);
                finish = indeces{i}(j+1)-1;
                n = round((finish - start) / add_spike_scale);
                [peak, k] = min(current{i}(start:finish));
                timepeak = repmat(time{i}(start+k), n, 1);
                time{i} = [time{i}(1:start+k-1); timepeak;...
                                                   time{i}(start+k+1:end)];
                current{i} = [current{i}(1:start+k-2);...
                              repmat(peak, n, 1); current{i}(start+k:end)];
                indeces{i}(j+1:end) = indeces{i}(j+1:end) + n - 1;
            end
        end
    end
end

% median pulse index
if nonlinfit
    mpi = ceil(length(pulses) / 2);
    hack = @(c, i) c{i};
    nlinfit_func = @(beta, time) hack(fit_function(beta, time,...
                                                          pulses(mpi)), 1);
    nlinfit_time = time(mpi);
    opt = statset('MaxIter', 1000);
    [beta_nlfit, res, J] = nlinfit(nlinfit_time, current{mpi},...
                                                 nlinfit_func, beta0, opt);
else
    switch transporter
        case 'e1'
           beta_nlfit = [940.28, 61.40, 56.39, 244.44, 90.71, 12.82,...
                           12.87, 856.33, 1232.35, 733.6];
        case 'e2'
            beta_nlfit = [1000, 0, 100, 10, 100, 100, 100, 100, 100, 0];
        case 'e3'
            beta_nlfit = [1041, 1, 110, 104, 289, 220, 61, 669, 1343, 665];
        otherwise
            beta_nlfit = beta0;
    end
end

if ~run_mcmc
    ci = nlparci(beta_nlfit, res, 'jacobian', J);
    figure(1)
        plot(nlinfit_time{1}, current{mpi}, 'k',...
             nlinfit_time{1}, nlinfit_func(beta_nlfit, nlinfit_time), 'r')
else
    data.time = time;
    data.current = current;
    data.indeces = indeces;
    data.pulses = pulses;

    beta_mcmc = beta_nlfit;
    model.ssfun = @ssfun;
    model.N = sum(cellfun(@(c) c(end), indeces));
    model.sigma2 = ssfun(beta_mcmc, data) / (model.N - length(beta_mcmc));

    params = {
    {'m_1^+', abs(beta_mcmc(1)), 0}
    {'m_1^-', abs(beta_mcmc(2)), 0}
    {'m_2^+', abs(beta_mcmc(3)), 0}
    {'m_2^-', abs(beta_mcmc(4)), 0}
    {'m_3^+', abs(beta_mcmc(5)), 0}
    {'m_4^+', abs(beta_mcmc(6)), 0}
    {'m_4^-', abs(beta_mcmc(7)), 0}
    };
    i = 1;
    k = length(params) - 3;
    params{k + 4 * i} = {sprintf('A%i', i), beta_mcmc(8)};
    params{k + 1 + 4 * i} = {sprintf('B%i', i), beta_mcmc(9)};
    params{k + 2 + 4 * i} = {sprintf('C%i', i), beta_mcmc(10)};

    options.nsimu = nsimu;
    options.qcov = 1000 * eye(length(beta_mcmc));
    options.method = 'dram';
    options.adaptint = 100;
    options.updatesigma = 0;
    options.drscale = [10 100];
    options.ntry = length(options.drscale) + 1;
    options.verbosity = 1;
    options.waitbar = 1;
    options.burnintime = 30000;

    [results,chain,s2chain] = mcmcrun(model,data,params,options);

    I = fit_function(results.theta, time, pulses);
    figure(100), hold on
        for i = 1:length(I)
```

```
            plot(time{i}(indeces{i}(1):indeces{i}(end)),...
                    data.current{i}(indeces{i}(1):indeces{i}(end)), 'k',...
                    time{i}(indeces{i}(1):indeces{i}(end)), I{i}, 'g')
        end
        xlabel('s')
        ylabel('pA')
        hold off
    figure(200)
        mcmcplot(chain,1:7,results,'pairs',2);

    rate_chain = zeros(size(chain, 1), 1);
    rate_chain = to_rate(chain);
    ci = prctile(rate_chain, [0.5, 2.5, 97.5, 99.5, 50]);
    fprintf(['Rate: %0.5f per second\nCI 95%%: [%0.2f, %0.2f]\n',...
           'CI 99%%: [%0.2f, %0.2f]\n'], ci(5), ci(2), ci(3), ci(1), ci(4))
    figure(300)
        hist(rate_chain, 20), hold on
        plot([ci(2), ci(3)], [0, 0], 'r.', 'MarkerSize', 30)
        plot([ci(1), ci(4)], [0, 0], 'r*', 'MarkerSize', 10), hold off
        xlabel('Turnover rate (s^{-1})', 'FontSize', 12)
   save(['optimized_', transporter, '_10p.mat'], 'results', 'chain',...
                                                    's2chain', 'rate_chain')
   'finished'
   for i = 1:10
       figure(i)
       plot(chain(:, i))
   end
end
```

EAAT/fit_function.m

```
function I = fit_function(beta, t, pulses)
    I = cell(size(pulses));
    beta(1:7) = abs(beta(1:7));
    bta = num2cell(beta);
    % number of common parameters
    ncp = 10;
    % number of individual paramaters
    nip = 0;

    for i = 1:length(pulses)
        ind = [1:ncp, (ncp + (1 + (i - 1) * nip:i * nip))];
        [m1p, m1m, m2p, m2m, m3p, m4p, m4m, A, B, C] = deal(bta{ind});
        % individual beta
        ibeta = [m1p, m1m, m2p, m2m, m3p, m4p, m4m, A, B, C];
        y = solve_system(ibeta, t{i}, pulses{i});
        I{i} = - A * y(:, 1) - B * y(:, 2) - C * y(:, 4) +...
               (A * m4p + C * m4m) / (m4p + m4m);
    end
```

```
end
```

EAAT/solve_system.m

```
function y = solve_system(beta, time, pulses)
    bta = num2cell(beta);
    [m1p, m1m, m2p, m2m, m3p, m4p, m4m] = deal(bta{1:7});
    y = zeros(length(time), 4);

    % steady state
    t = time(time >= pulses(1) & time <= pulses(2));
    y(1:length(t), :) = repmat([m4p, 0, 0, m4m] / (m4m + m4p),...
                                                             length(t), 1);

    % for all consecuitive pulses
    i = 3;
    j = length(t);
    A_template = @(m1pp)...
                  [-(m1pp + m4m), m1m, 0, m4p;
                   m1pp, -(m1m + m2p), m2m, 0;
                   0, m2p, -(m2m + m3p), 0;
                   m4m, 0, m3p, -m4p];
    A_pulse = A_template(m1p);
    A_washout = A_template(0);
    [V_p, D_p] = eig(A_pulse);
    [V_w, D_w] = eig(A_washout);

    while i <= length(pulses)
        y0 = y(j, :)';
        t = time(time > pulses(i - 1) & time <= pulses(i));
        t = t - t(1);
        if mod(i, 2) == 1
            % pulse of glutamate
            V = V_p;
            E = diag(D_p);
        else
            % washout
            V = V_w;
            E = diag(D_w);
        end
        C = V \ y0;
        y(j+1:j+length(t), :) = real((repmat(C, 1, length(t))...
                                               .* exp(kron(E, t')))' * V');
        j = j + length(t);
        i = i + 1;
    end
    y = y(1:j, :);
end
```

EAAT/ssfun.m

```
function y = ssfun(beta, data)
    I = fit_function(beta, data.time, data.pulses);
    y = 0;
    ind = data.indeces;
    for i = 1:length(I)
        y = y + sum((data.current{i}(ind{i}(1):ind{i}(end)) - I{i}).^2);
    end
end
```

EAAT/to_rate.m

```
function rate = to_rate(beta)
    rate = zeros(size(beta, 1), 1);
    for i = 1:size(beta, 1)
        bta = num2cell(beta(i, :));
        [m1p, m1m, m2p, m2m, m3p, m4p, m4m] = deal(bta{1:7});
        rate(i) = m1p * m2p * m3p * m4p / (m1p * m2p * (m3p + m4p) +...
                                        m1p * m4p * (m2m + m3p) +...
              (m4p + m4m) * (m1m * m2m + m1m * m3p + m2p * m3p));
    end
end
```

B.2 Synaptic diffusion model

Synapse/run.m

```
clear all

%% Parameters
% Space grid size
I = 30;
J = 31;
K = 4;
% Number of pulses
Cmax = 1;
% Delay between pulses (ms). 100Hz ~ 10ms
Td = 10;
% Wait after the last pulse (ms)
Tw = 1000;
% Astrocyte coverage
coverage = 0.95;
% draw a figure?
draw = false;
% save?
save_data = true;

%% Constants (in nm, ms, #1e-6molecules)
% numerical modifier for molecules
scaling = 1e6;
% ambient glutamate (25nM)
u_inf = @(t) 25 * 6.022e-10 * scaling;
% Height of a synaptic cleft (20nm)
L = 20;
% Radius of a synaptic cleft (150nm)
R = 150;
% Radius of an active zone (100nm)
ra = 100;
% Radius of an postsynaptic density (100nm)
rd = 100;
% Diffusion constant (0.4e-9 m^2 / s = 0.4 um^2 / ms)
D = 0.4e6;
% EAATs densities (1, 2a, 2n, 3) [2300 7500 750 90] um^-2
N = [2300 7500 750 90] * 1e-6;
% EAATs turnover rates (1 - 3) [15.45 23.96 1.98] s^-1
V = [15.45, 23.96, 1.98] * 1e-3 * scaling;
% EAATs Km's (1 - 3) [20, 18, 28] uM
Km = [20, 18, 28] * 6.022e-7 * scaling;
% pulse apmlitude (delta function) molecules
delta = 3000 * scaling;
```

```
%% Define system of ODE
% step sizes, indeces, matrices, etc.
dr = R / I;
dp = 2 * pi / (J + 1);
dz = L / K;
params = make_matrices(dr, dp, dz, I, J, K, [9, 17, 25], coverage);
% params.dt = dt;
params.ia = round(ra / dr) + 1;
params.id = round(rd / dr) + 1;
params.D = D;
params.N = N;
params.V = V;
params.Km = Km;
params.u_inf = u_inf;
% rhs
rhs = @(t, y) rhs_template(t, y, params);
% Initial conditions before the first pulse
y0 = u_inf(0) * ones(K + 1 + I * (J + 1) * (K + 1), 1);

if draw
    f1 = figure(1);
    f1.Position = [10 90 1400 900];
end
[rgrid, phigrid] = meshgrid(0:dr:R, 0:dp:2*pi-dp);

yc = cell(1, Cmax);
time = {};
tc = 0;
for cycle = 1:Cmax
    % 1/4 * h * S_base = delta / 2
    y0(1, 1) = y0(1, 1) + delta * 4 / (pi * dz * dr^2);
    switch cycle
        case Cmax
            tc = tc(end) + [0 Tw];
        case 1
            tc = [0 Td];
        otherwise
            tc = tc(end) + [0 Td];
    end
    [time{cycle}, y] = ode15s(rhs, tc, y0);
    % initial condition for the next pulse
    y0 = y(end, :)';
    if draw
        for i = 2:size(y, 1)
            x = y(i, :)';
            u = zeros(I + 1, J + 1, K + 1);
            u(1, :, :) = repmat(reshape(x(1:K+1), [1, 1, K + 1]), 1, J + 1, 1);
            u(2:end, :, :) = reshape(x(K + 2:end), [I, J + 1, K + 1]);
            for j = 1:4
                subplot(2, 2, j)
                surf(rgrid .* cos(phigrid), rgrid .* sin(phigrid),...
                     u(:, :, j)'/scaling);
                view(0.1, 90)
                set(gca, 'colorscale', 'log')
                caxis([1e-10 1e0])
                colorbar()
            end
            suptitle(sprintf('%s ms', time{cycle}(i)))
            pause(0.1)
        end
    end
    yc{cycle} = y;
end


% Concentration at the center and average at psd,
% and astrocytic flux
cca = [];
% flx = [];
total_time = [];
for i = 1:length(yc)
    for t = 1:size(yc{i}, 1)
        x = yc{i}(t, :)';
        u = zeros(I + 1, J + 1, K + 1);
        u(1, :, :) = repmat(reshape(x(1:K + 1), [1, 1, K + 1]), 1, J + 1, 1);
        u(2:end, :, :) = reshape(x(K + 2:end), [I, J + 1, K + 1]);
        w = u(1:params.id, :, end);
        rw = rgrid(1, 1:params.id);
        f = @(r) r .* interp1(rw, mean(w, 2), r);
        cca = [cca; integral(f, rw(1), rw(end)) * 2 / rw(end)^2];
    end
    total_time = [total_time; time{i}];
end

f2 = figure(2);
f2.Position = [424 319 797 659];
    plot(total_time, cca/6.022e-4/scaling, 'k')% from #molecules/nm^3 to mM

if save_data
    file_name_to_save = sprintf('%d_%d_at_%dHz.mat', coverage, Cmax, 1000/Td);
    save(file_name_to_save, 'time', 'yc', 'params', 'total_time', 'cc',...
                                                         'cca', 'scaling')
end
```

Synapse/make_matrices.m

```
function p = make_matrices(dr, dp, dz, I, J, K, ipg, coverage)
    alpha = [-4, - 2 * ones(1, I)] / dr^2;
    beta = [0, (1 + 0.5 ./ (1:I))] / dr^2;
    gamma = (1 - 0.5 ./ (0:I)) / dr^2;

    p.A = diag(alpha) + diag(beta(1:end-1), 1) + diag(gamma(2:end), -1);
    p.B = (- 2 * diag(ones(1, J + 1)) + diag(ones(1, J), 1) +...
        diag(ones(1, J), -1)) / dp^2;
    p.BB = repmat([0; 1 ./ (1:I)'.^2], 1, J + 1) / dr^2;
    p.C = (- 2 * diag(ones(1, K + 1)) + diag(ones(1, K), 1) +...
        diag(ones(1, K), -1)) / dz^2;

    p.beta_I = beta(end);
    p.gamma_0 = gamma(1);
    p.I = I;
    p.J = J;
    p.K = K;
    p.dr = dr;
    p.dp = dp;
    p.dz = dz;
    p.ipg = ipg;
    p.jc = round(2 * pi * coverage / dp);
end
```

Synapse/rhs_template.m

```
function y = rhs_template(t, x, p)
    u = zeros(p.I + 1, p.J + 1, p.K + 1);
    u(1, :, :) = repmat(reshape(x(1:p.K + 1), [1, 1, p.K + 1]), 1, p.J + 1, 1);
    u(2:end, :, :) = reshape(x(p.K + 2:end), [p.I, p.J + 1, p.K + 1]);

    MM = @(u, Km) u ./ (Km + u);
    E = zeros(p.I + 1, p.J + 1, p.K + 1);
    E(1, :, :) = repmat(u(2, 1, :) + u(2, p.ipg(1), :) +...
                        u(2, p.ipg(2), :) + u(2, p.ipg(3), :), 1, p.J + 1, 1)...
                        / p.dr^2;

    J2 = p.N(3) * p.V(2) * (MM(u(p.ia + 1:end, :, 1), p.Km(2)) -...
                             MM(p.u_inf(t), p.Km(2)));
    J3 = p.N(4) * p.V(3) * (MM(u(p.id + 1:end, :, end), p.Km(3)) -...
                             MM(p.u_inf(t), p.Km(3)));
    J12 = p.N(1) * p.V(1) * (MM(u(end, 1:p.jc, :), p.Km(1)) -...
                             MM(p.u_inf(t), p.Km(1)))+...
          p.N(2) * p.V(2) * (MM(u(end, 1:p.jc, :), p.Km(2)) -...
                             MM(p.u_inf(t), p.Km(2)));

    E(end, 1:p.jc, :) = (- J12 * 2 * p.dr / p.D + u(end - 1, 1:p.jc, :))...
                            * p.beta_I;
    E(end, p.jc+1:end, :) = ones(1, p.J + 1 - p.jc, p.K+1)...
                            * p.u_inf(t) * p.beta_I;

    E(2:end, 1, :) = E(2:end, 1, :) + u(2:end, end, :) / p.dp^2 /...
```

```
                            p.dr^2 ./ repmat(((1:p.I)').^2, 1, 1, p.K + 1);
    E(2:end, end, :) = E(2:end, end, :) + u(2:end, 1, :) / p.dp^2 /...
                            p.dr^2 ./ repmat(((1:p.I)').^2, 1, 1, p.K + 1);

    E(1:p.ia, :, 1) = E(1:p.ia, :, 1) + u(1:p.ia, :, 2) / p.dz^2;
    E(p.ia + 1:end, :, 1) = E(p.ia + 1:end, :, 1) +...
                  (- J2 * 2 * p.dz / p.D + u(p.ia + 1:end, :, 2)) / p.dz^2;
    E(1:p.id, :, end) = E(1:p.id, :, end) +...
                        u(1:p.id, :, end - 1) / p.dz^2;
    E(p.id + 1:end, :, end) = E(p.id + 1:end, :, end) +...
            (- J3 * 2 * p.dz / p.D + u(p.id + 1:end, :, end - 1)) / p.dz^2;

    y = zeros(size(E));
    for i = 1:p.K + 1
        y(:, :, i) = p.A * u(:, :, i) + (u(:, :, i) * p.B) .* p.BB +...
                      E(:, :, i);
    end
    for i = 1:p.J + 1
        y(:, i, :) = y(:, i, :) +...
                     reshape(reshape(u(:, i, :), [p.I+1, p.K+1]) * p.C,...
                                                    [p.I+1, 1, p.K+1]);
    end
    y = [reshape(y(1, 1, :), p.K+1, 1);...
         reshape(y(2:end, :, :), p.I * (p.J + 1) * (p.K + 1), 1)] *...
         p.D;
end
```

B.3 Glutamate receptor model

NMDAR/run.m

```
%% Define true parameters (model B)
% on (uM^-1 s^-1); off (s^-1)
k1_on = 4.0;   k1_off = 0.97;
k2_on =  1.7;  k2_off = 2.35;
k3_on = 4.0;   k3_off = 8.25;
k4_on = 1.7;   k4_off = 19.9;
k5_on = 3.68;  k5_off = 3.0;
k6_on = 83.8;  k6_off = 83.8;
params = [k2_on; k2_off; k1_on; k1_off; k2_on; k2_off; k1_on; k1_off;...
          k2_on; k2_off; k1_on; k1_off; k2_on; k2_off; k1_on; k1_off;...
          k2_on; k2_off; k1_on; k1_off; k4_on; k4_off; k3_on; k3_off;...
          k5_on; k5_off; k6_on; k6_off];

% time course and substrate concentrations
time = {};
time{1} = 0:0.1:10;
time{2} = max(time{1}):0.001:(max(time{1})+5);
time{3} = max(time{2}):0.001:(max(time{2})+3.5);

% concentrations (uM)
Smin = 10; Smax = 10000;
Gmin = 10; Gmax = 10000;

% current scale factor (nA (normalized transporter concentration)^-1 )
% and noise level
C = -10;
eps = 0.01;

%% Simulate data
simulated_data = sim_data(time, params, Gmin, Gmax, Smin, Smax, C, eps);

%% Model fitting
k1_on = 1.7;    k1_off = 2.35;
k2_on = 4.0;    k2_off = 0.97;
k3_on = 1.7;    k3_off = 2.35;
k4_on = 4.0;    k4_off = 0.97;
k5_on = 1.7;    k5_off = 2.35;
k6_on = 4.0;    k6_off = 0.97;
k7_on = 1.7;    k7_off = 2.35;
k8_on = 4.0;    k8_off = 0.97;
k9_on = 1.7;    k9_off = 2.35;
k10_on = 4.0;   k10_off = 0.97;
k11_on = 1.7;   k11_off = 19.9;
k12_on = 4.0;   k12_off = 8.25;
k13_on = 3.68;  k13_off = 3.0;
```

```
k14_on = 83.8;  k14_off = 83.8;
C = -10;
% params0 = [k1_on, k1_off, k2_on, k2_off, k3_on, k3_off, k4_on, k4_off,...
%       k5_on, k5_off, k6_on, k6_off, k7_on, k7_off, k8_on, k8_off,...
%       k9_on, k9_off, k10_on, k10_off, k11_on, k11_off, k12_on, k12_off,...
%       k13_on, k13_off, k14_on, k14_off];
% params0 = params0 + eps * randn(size(params0));

%% Fitting
stage2 = (length(time{1})+1):(length(time{1})+length(time{2}));
stage3 = (length(time{1})+length(time{2})+1):...
                         (length(time{1})+length(time{2})+length(time{3}));
% Step 1. Fit (K13, K14, C)
step1_params0 = [k13_on, k13_off, k14_on, k14_off, C];
step1_params0 = step1_params0 + 0.1 * randn(size(step1_params0));

[step1_fit, step1_res, step1_J] = nlinfit(time, simulated_data{1}(stage2),...
                                            @step1_fit_func, step1_params0);
step1_fit_ci = nlparci(step1_fit, step1_res, 'jacobian', step1_J);

% Step 2. Fit (K6, K12)
step2_params0 = [k6_on, k6_off, k12_on, k12_off];
step2_params0 = step2_params0 + 0.1 * randn(size(step2_params0));

[step2_fit, step2_res, step2_J] = nlinfit(time, [simulated_data{3}(stage2);...
               simulated_data{3}(stage3); simulated_data{1}(stage3)],...
             @(b, t) step23_fit_func(b, t, step1_fit, Gmin), step2_params0);
step2_fit_ci = nlparci(step2_fit, step2_res, 'jacobian', step2_J);

% Step 3. Fit (K9, K11)
step3_params0 = [k9_on, k9_off, k11_on, k11_off];
step3_params0 = step3_params0 + 0.1 * randn(size(step3_params0));

[step3_fit, step3_res, step3_J] = nlinfit(time, [simulated_data{6}(stage2);...
               simulated_data{6}(stage3); simulated_data{2}(stage3)],...
             @(b, t) step23_fit_func(b, t, step1_fit, Smin), step3_params0);
step3_fit_ci = nlparci(step3_fit, step3_res, 'jacobian', step3_J);

% Step 4. (K1, K3)
step4_params0 = [k1_on / k1_off, k3_on / k3_off];
step4_params0 = step4_params0 + 0.1 * randn(size(step4_params0));

[step4_fit, step4_res, step4_J] = nlinfit(time, simulated_data{5}(stage2),...
             @(b, t) step45_fit_func(b, t, [step3_fit, step1_fit], Smin),...
                                                            step4_params0);
step4_fit_ci = nlparci(step4_fit, step4_res, 'jacobian', step4_J);

% Step 5. (K2, K8)
step5_params0 = [k2_on / k2_off, k8_on / k8_off];
step5_params0 = step5_params0 + 0.1 * randn(size(step5_params0));
```

```
[step5_fit, step5_res, step5_J] = nlinfit(time, simulated_data{4}(stage2),...
             @(b, t) step45_fit_func(b, t, [step2_fit, step1_fit], Gmin),...
                                                             step5_params0);
step5_fit_ci = nlparci(step5_fit, step5_res, 'jacobian', step5_J);
```

NMDAR/sim_data.m

```
function current = sim_data(time, params, Gmin, Gmax, Smin, Smax, C, eps)
    current = {};
    % High concentrations
    %       Type 1
    current{1} = sim_exp(time, params, @true_rhs, Gmax, Smax, C, 1, eps);
    %       Type 2
    current{2} = sim_exp(time, params, @true_rhs, Gmax, Smax, C, 2, eps);

    % High Serine, low Glutamate
    %       Type 1
    current{3} = sim_exp(time, params, @true_rhs, Gmin, Smax, C, 1, eps);
    %       Type 2
    current{4} = sim_exp(time, params, @true_rhs, Gmin, Smax, C, 2, eps);

    % High Glutamate, low Serine
    %       Type 1
    current{5} = sim_exp(time, params, @true_rhs, Gmax, Smin, C, 1, eps);
    %       Type 2
    current{6} = sim_exp(time, params, @true_rhs, Gmax, Smin, C, 2, eps);

    % Low Glutamate, low Serine
    %       Type 1
    current{7} = sim_exp(time, params, @true_rhs, Gmin, Smin, C, 1, eps);
    %       Type 2
    current{8} = sim_exp(time, params, @true_rhs, Gmin, Smin, C, 2, eps);
end
```

NMDAR/sim_exp.m

```
function current = sim_exp(time, params, rhs, G, S, C, T, eps)
    y0 = [1; zeros(10, 1)];
    x = {};
    if T == 1
        p = [params; 0; S];
    else
        p = [params; G; 0];
    end
    [~, x{1}] = ode23s(@(t, x) rhs(t, x, p), time{1}, y0);
    [~, x{2}] = ode23s(@(t, x) rhs(t, x, [params; G; S]), time{2},...
                                            x{1}(end, :));
    [~, x{3}] = ode23s(@(t, x) rhs(t, x, p), time{3}, x{2}(end, :));

    current = [x{1}(:, 11); x{2}(:, 11); x{3}(:, 11)];
    current = C * current + eps * randn(size(current));
end
```

NMDAR/true_rhs.m

```
function y = true_rhs(~, x, params)
    params = num2cell(params);
    [k1_on, k1_off, k2_on, k2_off, k3_on, k3_off, k4_on, k4_off,...
        k5_on, k5_off, k6_on, k6_off, k7_on, k7_off, k8_on, k8_off,...
        k9_on, k9_off, k10_on, k10_off, k11_on, k11_off,...
        k12_on, k12_off, k13_on, k13_off, k14_on, k14_off, G, S] =...
                                                           deal(params{:});
    x = num2cell(x);
    [a, eta, b, x, th, g, y, xi, z, zp, zs] = deal(x{:});
    y = [ % alpha
         (- k1_on * a * S + k1_off * eta - k2_on * a * G + k2_off * b);
          % eta
         (k1_on * a * S - k1_off * eta - k3_on * eta * S + k3_off * x...
         - k4_on * eta * G + k4_off * th);
          % beta
         (k2_on * a * G - k2_off * b - k5_on * b * S + k5_off * th...
         -k8_on * b * G + k8_off * g);
          % x
         (k3_on * eta * S - k3_off * x - k6_on * x * G + k6_off * y);
          % theta
         (k4_on * eta * G - k4_off * th + k5_on * b * S - k5_off * th...
         - k7_on * th * S + k7_off * y - k10_on * th * G + k10_off * xi);
         % gamma
         (k8_on * b * G - k8_off * g - k9_on * g * S + k9_off * xi);
         % y
         (k6_on * x * G - k6_off * y + k7_on * th * S...
         - k7_off * y - k12_on * y * G + k12_off * z);
         % xi
         ( k9_on * g * S - k9_off * xi + k10_on * th * G - k10_off * xi...
         - k11_on * xi * S + k11_off * z);
         % z
         (k11_on * xi * S - k11_off * z + k12_on * y * G - k12_off * z...
         - k13_on * z + k13_off * zp - k14_on * z + k14_off * zs);
         % zp
         (k13_on * z - k13_off * zp);
         % zs
         (k14_on * z - k14_off * zs)];
end
```

NMDAR/hone_rhs.m

```
function y = hone_rhs(~, x, params)
    params = num2cell(params);
    % kX and kY are:
    %                  k6 and k12 for high Serine and low Glutamate
    %                  k9 and k11 for high Glutamate and low Serine
    % L is:
    %                  G for high Serine and low Glutamate
    %                  S for high Glutamate and low Serine
    [kX_on, kX_off, kY_on, kY_off, k13_on, k13_off, k14_on, k14_off,...
     L] = deal(params{:});
    x = num2cell(x);
    % X and Y are:
    %                  RS2 and GRS2 for high Serine and low Glutamate
    %                  RG2 and G2RS for high Glutamate and low Serine
    [X, Y, z, zp, zs] = deal(x{:});
    y = [% RS2 / RG2
         (- kX_on * X * L + kX_off * Y);
         % GRS2 / G2RS
         (kX_on * X * L - kX_off * Y - kY_on * Y * L + kY_off * z);
         % G2RS2
         (kY_on * Y * L - kY_off * z - k13_on * z + k13_off * zp...
         - k14_on * z + k14_off * zs);
         % G2RpS2
         (k13_on * z - k13_off * zp);
         % G2RsS2
         (k14_on * z - k14_off * zs)];
end
```

NMDAR/hboth_rhs.m

```
function y = hboth_rhs(~, x, params)
    params = num2cell(params);
    [k13_on, k13_off, k14_on, k14_off] = deal(params{:});
    x = num2cell(x);
    [z, zp, zs] = deal(x{:});
    y = [% z
         (- k13_on * z + k13_off * zp...
         - k14_on * z + k14_off * zs);
         % zp
         (k13_on * z - k13_off * zp);
         % zs
         (k14_on * z - k14_off * zs)];
end
```

NMDAR/step1_fit_func.m

```
function fit_data = step1_fit_func(params, time)
    [~, x] = ode23s(@(t, x) hboth_rhs(t, x, params(1:end-1)), time{2},...
                                                            [1; 0; 0]);
    fit_data = params(end) * x(:, end);
end
```

NMDAR/step23_fit_func.m

```
function fit_data = step23_fit_func(params, time, known, L)
    [~, x1] = ode23s(@(t, x) hone_rhs(t, x, [params, known(1:end-1), L]),...
                time{2}, [1; 0; 0; 0; 0]);
    [~, x2] = ode23s(@(t, x) hone_rhs(t, x, [params, known(1:end-1), 0]),...
                time{3}, x1(end, :)');
    [~, y] = ode23s(@(t, x) hboth_rhs(t, x, known(1:end-1)), time{2},...
                                                            [1; 0; 0]);
    [~, x3] = ode23s(@(t, x) hone_rhs(t, x, [params, known(1:end-1), 0]),...
                time{3}, [0; 0; y(end, :)']);
    fit_data = known(end) * [x1(:, end); x2(:, end); x3(:, end)];
end
```

NMDAR/step45_fit_func.m

```
function fit_data = step45_fit_func(params, time, known, L)
    [~, x] = ode23s(@(t, x) hone_rhs(t, x, [known(1:end-1), L]),...
        time{2}, [1; params(1) * L; params(1) * params(2) * L^2; 0; 0]./...
                          (1+ params(1) * L + params(1) * params(2) * L^2));
    fit_data = known(end) * x(:, end);
end
```