# Theory on the accurate estimation of Michaelis-Menten enzyme kinetic parameters from steady state and progress curve datasets

R Murugan

Department of Biotechnology, Indian Institute of Technology Madras
Chennai, India. Email: rmurugan@gmail.com

## ABSTRACT

Accurate estimation of enzyme kinetic parameters such as $K_M$ and $v_{max}$ from the experimental datasets is critical to characterize any enzyme. The validity of widely used standard quasi steady state approximation over decades relies on *a priori* information of $K_M$ and other kinetic parameters which are generally not available for newly identified enzyme systems. Progress curve analysis of the time course datasets was proposed to alleviate such issues. Since steady state and progress curve analysis use different types of datasets, the estimated $K_M$ values will be inconsistent among these methods. We show that the error in the estimation of $K_M$ using steady state methods is strongly dependent on the total reaction time. Progress curve methods work well at the timescale regime with maximum change in the curvature of the trajectory of substrate evolution. Whereas, steady state methods work across the timescales with minimum change in the curvature of the trajectory of product evolution. We propose an integrated approach which comprises of progress curve and three different steady state methods using the same time course dataset that also consider the total reaction time. We demonstrate that there exists an optimum reaction time at which all the steady state and progress curve methods show minimum possible error in the estimation of $K_M$. We define the reliability index as the coefficient of variation of the median $K_M$ values obtained from progress and various steady state methods using the same datasets. Accurate estimate of $K_M$ can be obtained by iterating the total reaction time towards the least possible reliability index.



## KEY WORDS

Michaelis-Menten enzyme kinetics, $K_M$, enzyme kinetic parameters, accurate estimation, nonlinear least square fitting.

## Introduction

Enzymes catalyse several crucial biochemical reactions especially at physiologically relevant timescales and environmental conditions [1-3]. The mechanistic view of single substrate enzyme catalysis can be well described by the Michaelis-Menten scheme (MMS) [4, 5]. In this scheme (**Fig. 1A**), the enzyme first binds the substrate to form an enzyme-substrate complex which reduces the activation energy barrier of substrate to product conversion. Subsequently, the substrate will be converted into the respective product and released. The MMS scheme is characterized by three different rate constants viz. diffusion-controlled bimolecular forward $k_1$ (1/mol/lit/second), unimolecular reverse $k_{-1}$ (1/second) and product formation rate constants $k_2$ (1/second). These rate constants along with the initial enzyme level ($e_0$) are generally summarized by two important enzyme kinetic parameters viz. $K_M = (k_{-1} + k_2) / k_1$ (mol/lit) and $v_{max} = k_2\, e_0$ (mol/lit/s). Here $k_2$ is also termed as $k_{cat}$ and the ratio $k_E = k_{cat} / K_M$ is defined as the catalytic efficiency of the enzyme. When $k_2 \gg k_{-1}$, then $k_E$ reaches the diffusion-controlled rate limit $k_1$.

Exact closed form analytical solution to the nonlinear set of ODEs corresponding to MMS scheme is not known though the integral solution can be expanded in terms of ordinary [6] and singular perturbation series [7] or over slow manifolds [8, 9]. Singular perturbation expansions always yield inner and outer solutions corresponding to the pre-steady state and post-steady state regimes which should be combined via proper matching at the temporal boundary layer [7, 10-15] to obtain the complete solution. Whereas, ordinary perturbation series can give the complete solution that is valid for both pre- and post-steady state regimes [6]. Obtaining ($K_M$, $k_{cat}$, $v_{max}$) from the steady state and progress curve experimental datasets is essential to characterize an enzyme. Since complete closed form analytical solution to MMS is unknown, one needs to rely on the widely used standard quasi steady state approximation (sQSSA) of the reaction velocity $v = dp/dt$ as $v = v_{max}\, s / (K_M + s)$ that is valid only when $k_2 \ll (k_1\, s_0)$ [16]. Here p represents the product concentration. When $e_0 \ll s_0$, then one can use the stationary reactant assumption to further simplify as $v \cong v_0 = v_{max}\, s_0 / (K_M + s_0)$ which connects $K_M$, $v_{max}$ and initial substrate concentration $s_0$. Nonlinear least square fitting of the data on ($v$, $s_0$) over original hyperbolic function or linearization techniques such as Lineweaver-Burk on ($1/v$, $1/s_0$) and Eadie-Hofstee on ($v$, $v/s_0$) [17] can be used to obtain the estimates of kinetic parameters [18, 19]. Several statistical analysis procedures were also proposed [20, 21] to compute the error on these parameter estimates.

Enzymes of pathogenic organisms are important targets for drug molecules to control various diseases. Accurate estimation of various MMS parameters especially $K_M$ along with $K_I$ of a competitive inhibitor is essential to design and screen various drug molecules against a given target enzyme [22]. In steady state experiments, a fixed amount of enzyme will be incubated with a series of different substrate concentrations ($s_0$) for fixed amount of time ($t_r$) at constant temperature and pH conditions from which the *average product formation velocity* ($\bar{v}$) will be calculated by the ratio = total product formed during incubation time / total incubation time ($t_r$). Under sQSSA conditions with $e_0 \ll s_0$, it will be approximately assumed that the system evolves with constant reaction velocity $v_0 = v_{max}\, s_0 / (K_M + s_0)$ after a short pre-steady state timescale $t_c \ll t_r$ so that the total product formed over time $t_r$ will be approximately equal to ($p = v_0\, t_r$) and therefore one sets the average reaction velocity $\bar{v} \cong \bar{v}_{sQSSA} = (p/t_r) \cong v_0$.

This equation is being used widely across enormous volumes of literature to estimate $K_M$ and $v_{max}$ from the dataset on ($\bar{v}$, $s_0$) obtained from steady state experiments. Here $\bar{v}$ is the average velocity = product formed in a given reaction time / total reaction time. Detailed studies proposed the essential and sufficient conditions for the validity of the approximation $\bar{v} \cong v_0$ viz. (a) $k_2 \ll (k_1 s_0)$ [16] (b) $e_0 \ll s_0$ [23, 24] (c) $(e_0 / (s_0 + K_M)) \ll 1$ [23, 24]. However, except (b), conditions (a) and (c) strictly require *a priori* knowledge on $k_1$, $k_2$ and $K_M$. Otherwise, one needs to set very high values of $s_0$ along with condition (b) so that all the conditions (a-c) are satisfied. Although the inequality (b) is possible under *in vitro* laboratory conditions, there are several situations such as single molecule enzyme kinetics [25] and other *in vivo* conditions where one cannot manipulate the ratio of substrate to enzyme concentrations much. Setting high $s_0$ along with infinitesimal $e_0 \ll s_0$ would eventually drives $\bar{v}$ towards infinitesimal values below the background noise level which results in noisy experimental data on ($\bar{v}$, $s_0$) that leads to enormous amount of error in the estimation of $K_M$. Several recent studies [26-29] reported this issue and also put forth alternate methods such as total QSSA (tQSSA) to obtain accurate $K_M$ values from such experimental datasets.

The source datasets for the estimation of $K_M$ will be either time dependent progress curve type [30] or steady state average velocity type. Here progress curve data on (p, s, t) contains information on both pre- as well as post steady state dynamics that in turn depends on the timescale of experimental data collection. Whereas, steady state experimental setups generally capture the details of only the post-steady state regime. Progress curve methods can yield accurate estimates of $K_M$ only when the data collection done at those timescale regimes with maximum change in the curvature of the kinetic trajectory [31]. Further, the role of total reaction time $t_r$ is not considered anywhere in the sQSSA calculations i.e., $\bar{v}_{sQSSA} \cong v_0$ though the final results strongly depend on $t_r$. In this context, integrated form of sQSSA (isQSSA) is more superior than the general sQSSA since it also considers $t_r$.

Since the conditions of validity of various methods like sQSSA, isQSSA and progress curve analysis are different from each other, $K_M$ values of the same enzyme obtained from each of these methods under identical conditions using the same source dataset will not be consistent. Further, absence of *a priori* knowledge on $K_M$ and other kinetic parameters will eventually introduces uncertainty in setting up the appropriate experimental conditions to minimize the error in estimation of these parameters using sQSSA methods with stationary reactant assumption or progress curve methods. In this article, we develop an integrated theoretical and computational approach which can produce consistent estimate of $K_M$ upon considering all these uncertainty aspects.

## Theoretical Methods

The nonlinear coupled differential rate equations corresponding to MM **Scheme A** shown in **Fig. 1** can be written as follows [16].

$$\frac{de}{dt} = -k_1 es + (k_{-1} + k_2)x \,. \qquad [1.1]$$

$$\frac{ds}{dt} = -k_1 es + k_{-1} x. \qquad [1.2]$$

$$\frac{dx}{dt} = k_1 es - (k_{-1} + k_2)x\,. \qquad [1.3]$$

$$\frac{dp}{dt} = v = k_2 x\,. \qquad [1.4]$$

Here $k_1$ (1/mol/lit/second), $k_{-1}$ (1/second) and $k_2$ (1/second) are the rate constants corresponding to bimolecular enzyme-substrate complex formation, dissociation and product formation respectively. Further, (*e*, *s*, *x*, *p*) (mol/lit) are the concentrations of enzyme, substrate, enzyme-substrate complex, product respectively and v (mol/lit/second) is the reaction velocity. The initial conditions at $t \to 0$ are $(e, s, x, v, p) = (e_0, s_0, 0,0,0)$ and the reaction trajectory ends at $(e, s, x, v, p) = (e_0, 0,0,0, s_0)$ as $t \to \infty$. The dynamical variables (*e*, *s*, *x*, *p*) obey the mass conservation laws viz. $s = (s_0 - x - p), e = (e_0 - x)$ where one can also replace $x = v/k_2$. Here the required enzyme kinetic parameters are defined as $K_M = (k_{-1} + k_2)/k_1$ and $v_{max} = k_2 e_0$. From the definition of $K_M$, one can conclude that accurate estimation of $K_M$ requires detailed data points representing both pre- and post-steady state dynamics. To simplify the system of **Eqs. 1,** we first introduce the following set of scaling transformations [16].

$$\tau = k_2 t;\ X = \frac{x}{e_0};\ P = \frac{p}{s_0};\ S = \frac{s}{s_0};\ E = \frac{e}{e_0};\ V = \frac{dP}{d\tau} \qquad [2]$$

In **Eqs. 2**, the variables (e, s, x, p) are normalized so that $0 \le (E, S, X, P) \le 1$ with the mass conservations laws $S = 1 - \varepsilon X + P$, $E + X = 1$. We define $V = \varepsilon X$ so that $V + P + S = 1$ where $\varepsilon = \left(\frac{e_0}{s_0}\right)$. **Eqs. 1** can be rescaled in multiple ways at different variable spaces by the following set of dimensionless nonlinear ODEs [16].

$$\eta \frac{d^2P}{d\tau^2} + \alpha \frac{dP}{d\tau} - \varepsilon(1 - P) = \frac{dP}{d\tau}\left(\frac{dP}{d\tau} + P\right). \qquad [3.1]$$

$$\eta \frac{dV}{d\tau} + \alpha V - \varepsilon(1 - P) = V(V + P);\ \frac{dP}{d\tau} = V. \qquad [3.2]$$

$$\eta \frac{dV}{d\tau} + \alpha V - \varepsilon(V + S) = V(1 - S);\ \eta \frac{dS}{d\tau} = (-(\varepsilon - V)S + \kappa V). \qquad [3.3]$$

$$(-(\varepsilon - V)S + \kappa V)\frac{dV}{dS} + \alpha V - \varepsilon(V + S) = V(1 - S). \qquad [3.4]$$

$$\eta V \frac{dV}{dP} + \alpha V - \varepsilon(1 - P) = V(V + P). \qquad [3.5]$$

$$\frac{dP}{dS} = -\frac{\eta(1-P-S)}{(\varepsilon-(1-P-S))S-\kappa(1-P-S)}\,. \qquad [3.6]$$

Here **Eq. 3.1** describes the MMS dynamics in (P, τ) space, **Eqs. 3.2** in (V, P, τ), **Eqs. 3.3** (V, S, τ), **Eq. 3.4** in (V, S), **Eq. 3.5** in (V, P) and **Eq. 3.6** in (P, S) spaces. Various dimensionless parameters in **Eqs. 3** are defined as follows.

$$\varepsilon = e_0/s_0\,;\ \eta = k_2/k_1 s_0\,;\ \kappa = k_{-1}/k_1 s_0\,;\ \mu = \eta + \kappa = K_M/s_0\,;\ \alpha = 1 + \varepsilon + \mu\,. \qquad [4]$$

When $\eta \to 0$, then **Eq. 3.4** results in sQSSA with the velocity $v_{sQSSA}$ in the original (v, s) space. Similarly, **Eq. 3.5** results in tQSSA with the velocity $v_{tQSSA}$ in the (v, p) space which can be written in terms of the original dynamical variables as follows.

$$v_{sQSSA} \cong \frac{v_{max}s}{K_M+s};\ v_{tQSSA} \cong \frac{1}{2s_0}\Big(K_M + e_0 + s_0 - p - s_0\sqrt{e_0^2 + 2e_0(p + K_M - s_0) + (p - K_M - s_0)^2}\Big). \quad [5]$$

We denote **Eqs. 5** as $\eta$-approximations since they are valid only when $\eta \to 0$. The main flaw of these sQSSA as well as tQSSA methods is that the left-hand sides of **Eqs. 5** are still time dependent quantities along with time dependent s and p on the right-hand sides. That is to say, $v_{sQSSA}$ and $v_{tQSSA}$ are tangents of progress curve in (p, t) space at a given time point and not the time averaged velocities as used in most of the steady state experiments. To derive equations which are useful to fit steady state experimental data, one needs to further set $\varepsilon \to 0$ in **Eqs. 5**. This leads to the stationary reactant assumption $s \cong s_0 - p \cong s_0$ that results in the constant velocity as $v_{sQSSA} \cong v_{tQSSA} \cong v_0 = \frac{v_{max}s_0}{K_M+s_0}$ since $lim_{\varepsilon\to 0} v_{tQSSA} \cong v_{sQSSA}$. In this scenario, one can conduct the average steady state velocity experiments only within the reaction timescales in which these conditions are valid.

Upon setting $\varepsilon \to 0$ in **Eq. 3.6**, one finds that $\frac{dP}{dS} \cong \frac{\eta}{S+\kappa}$ in the pre-steady state regime of (P, S) space. Solving this ODE for the initial condition P = 0 for S = 1, we obtain $P \cong \eta ln\left(\frac{S+\kappa}{1+\kappa}\right)$. Since $S \cong 1$ when $(\eta, \varepsilon) \to (0,0)$, one finds that $P \cong 0$ and $V \cong 1 - S$ in the pre-steady state regime ($p \cong 0$ and $v \cong k_2(s_0 - s)$ in terms of original variables). Since $s \cong s_0$ when $s_0 \gg e_0$, one finds that $v \cong 0$ for infinitesimal values of $k_2$ near the steady state regime. The appropriate working reaction timescales for the average steady state velocity experiments can be derived as follows. Using the transformation rule $P = \varepsilon\eta^2 U$, **Eq. 3.1** can be rewritten as the following ordinary perturbation equation with the scaling parameter $\phi = \eta\varepsilon$ as follows.

$$\frac{d^2U}{d\tau^2} + \frac{\alpha}{\eta}\frac{dU}{d\tau} + \frac{\varepsilon}{\eta}U - \frac{1}{\eta^3} = \phi\frac{dU}{d\tau}\left(\frac{dU}{d\tau} + U\right); U(0) = 0;\ U'(0) = 0. \quad [6]$$

When $\varepsilon \to 0$, then one arrives at the following approximations from **Eqs. 6** using the definition of dimensionless velocity $V = \varepsilon\eta^2 F$.

$$\frac{d^2U}{d\tau^2} + \frac{(1+\mu)}{\eta}\frac{dU}{d\tau} - \frac{1}{\eta^3} \cong 0;\ \frac{dU}{d\tau} = F;\ \frac{dF}{d\tau} + \frac{(1+\mu)}{\eta}F - \frac{1}{\eta^3} \cong 0; F(0) = 0\ . \quad [7]$$

We denote **Eqs. 7** as $\varepsilon$-approximations. Solving **Eqs. 7** for (U, F) in terms of τ and then reverting back to (P, V) and subsequently transforming back in to the original variables (p, v, t), one arrives at the following ε-approximations.

$$p_\varepsilon \cong (v_0 t - v_\varepsilon t_c) \cong v_0(t - t_c);\ v_\varepsilon \cong v_0\left(1 - exp\left(-\frac{t}{t_c}\right)\right);\ t_c \cong \frac{1}{k_1(K_M+s_0)}. \quad [8]$$

Here $v_0 = \frac{v_{max}s_0}{K_M+s_0}$ is the sQSSA with stationary reactant assumption i.e., $s \cong s_0$. When $t \gg t_c$, then one finds that $p_\varepsilon \cong v_0(t - t_c)$. From **Eqs. 8** one can also obtain $s_\varepsilon \cong s_0 - v_0 t - v_\varepsilon(1/k_2 - t_c)$ which follows from the mass conservation law $s_\varepsilon \cong s_0 - p_\varepsilon - v_\varepsilon/k_2$. The subscript ε in $p_\varepsilon$ and $s_\varepsilon$ corresponds to the ε-approximations. The pre-steady state timescale

in **Eq. 8** is $t_c$. When $t \gg t_c$, then one finds that $s_\varepsilon \cong \left(s_0 - v_0(1/k_2 - t_c)\right) - v_0 t$. When $t = t_c$, then the steady state concentration of substrate can be approximated as $s_{\varepsilon c} \cong s_0(1 - e_0/(K_M + s_0))$. Clearly, the stationary reactant assumption [32] $s_{\varepsilon c} \cong s_0$ will be true only when $(e_0/(K_M + s_0)) \ll 1$ which will be eventually true when $\varepsilon \to 0$. When the reaction time $t_r$ is such that $t_r < t_c$, then $p_\varepsilon \cong 0$ in the pre-steady state regime (infinitesimal quantity especially for $\varepsilon \to 0$). When $t_r \gg t_c$, then one finds the linear product evolution regime with reaction time $t_r$ as $p_\varepsilon \cong v_0(t_r - t_c)$ along with the constant reaction velocity $v_\varepsilon \cong v_0$. However, this linear dependence of $p_\varepsilon$ on reaction timescale will break down towards the end of reaction since the substrate concentration starts to decline towards zero beyond this timescale $t_m$. This approximately occurs when $t_r \geq t_m \cong \frac{s_0}{v_0} = \left(\frac{K_M + s_0}{k_2 e_0}\right)$. Since the product concentration evolves linearly with time as $v_0 t$ in the post-steady state regime, we find that $t_m \cong \frac{s_0}{v_0}$. These timescale behaviors are demonstrated using numerically computed integral trajectories of **Eqs. 1** in **Fig. 1B** and **1C**. Clearly, one finds that $t_m \propto s_0$, $lim_{s_0 \to \infty} t_c \to 0$, $lim_{s_0 \to \infty} t_m \to \infty$ and $lim_{s_0 \to \infty}\, v \to v_{max}$. Further, one can define the timescale separation ratio $\chi$ as follows.

$$\chi = \frac{t_m}{t_c} = \frac{k_1(K_M + s_0)^2}{k_2 e_0} = \frac{(1+\mu)^2}{\eta\varepsilon} = \frac{(1+\mu)^2}{\phi}. \qquad [9]$$

Here $\chi$ is inversely proportional to the perturbation parameter $\phi$. The successfulness of sQSSA methods strongly depends on extent of timescale separation that is decided by the condition $\chi \gg 1$ [16, 24] which also means that $\phi = \eta\varepsilon \ll (1+\mu)^2$. The linear dependence of product formation on time will be valid only when the reaction time falls in the range $t_c \ll t_r < t_m$ where one can conduct average reaction velocity experiments using $\bar{v} \cong \bar{v}_{sQSSA} = v_0$. However, fitting of steady state data on ($\bar{v}$, $s_0$) with $v_0$ will be biased towards $v_{max}$ over $K_M$ since there is not enough datapoints from the pre-steady state regime. When $t_r < t_m$, then one can also use **Eqs. 8** to conduct average velocity experiments in the following form.

$$\bar{v}_\varepsilon \cong \frac{p_\varepsilon}{t_r} \cong \frac{v_{max} s_0}{K_M + s_0}\left(1 - \frac{t_c}{t_r}\left(1 - exp\left(-\frac{t_r}{t_c}\right)\right)\right) = v_0\left(1 - \frac{t_c}{t_r}\left(1 - exp\left(-\frac{t_r}{t_c}\right)\right)\right). \qquad [10]$$

Here $p_\varepsilon$ is the total product formed at the end of reaction time $t_r$ as in **Eq. 8** and $\bar{v}_\varepsilon$ is the average velocity obtained from the steady state experiments. Unlike $\bar{v}_{sQSSA}$, $\bar{v}_\varepsilon$ also considers the nonlinearity of the pre-steady state regime. The conditions for the validity of **Eq. 10** are $\varepsilon \to 0$ and $t_r < t_m$. When $t_m > t_r \gg t_c$, then $\bar{v}_\varepsilon \to \bar{v}_{sQSSA} \cong v_0$. Here, $\bar{v}_{sQSSA} \cong v_0$ directly connects the average reaction velocity with the total substrate concentration $s_0$. Clearly, the conditions for the validity of the widely used $\bar{v}_{sQSSA} \cong v_0$ are viz. $(\eta, \varepsilon) \to (0,0)$ and $t_c \ll t_r < t_m$. Nonlinear least square fitting of the data on $(\bar{v}, s_0)$ for a given reaction time $t_r$ with **Eq. 10** can directly yield the parameters ($K_M$, $v_{max}$ and $k_1$). However, such fitting procedures favour accurate estimation of $v_{max}$ over ($K_M$, $k_1$) since they mainly rely on the datapoints from the pre-steady state timescale regime which is actually a transient one compared to the post-steady state timescale. To overcome this issue, we will fix $k_1$ in the definition of $\bar{v}_\varepsilon$ with the median $k_1$ obtained from the progress curve methods in the later sections. To minimize the error in the estimation of $K_M$ one needs to set $t_r$ such that it is much lesser than the maximum reaction time $t_m$ corresponding to the lowest $s_0$ value used in the entire experimental setup.

When $\eta \to 0$, then one finds as in **Eqs. 5** that $V \cong \frac{\varepsilon S}{\mu+S}$. When $\varepsilon \to 0$, then one finds that $\frac{dP}{d\tau} \cong -\frac{dS}{d\tau}$ in the post-steady state regime and therefore one obtains $-\frac{dS}{d\tau} \cong \frac{\varepsilon S}{\mu+S}$ (or $\frac{dP}{d\tau} \cong \frac{\varepsilon(1-P)}{\mu+(1-P)}$). Upon solving this nonlinear first order ODE with the initial condition S = 1 at $\tau = 0$, and then reverting back to the original dynamical variables, one finds the following implicit solution.

$$\bar{v}_{isQSSA} = \frac{s_0-s}{t_r} \cong \frac{p}{t_r} \cong v_{max} + \frac{K_M}{t_r} ln\left(\frac{s}{s_0}\right) \cong v_{max} + \frac{K_M}{t_r} ln\left(\frac{s_0-p}{s_0}\right). \quad [11]$$

**Eqs. 10** is another form of integrated rate equation [29] that directly connects the average reaction velocity $\bar{v}_{isQSSA}$ with the enzyme kinetic parameters and reaction time $t_r$. Data on $(\bar{v}, p, s, s_0)$ at a given reaction time $t_r$ can be directly used to perform linear least square fitting with **Eq. 11** i.e., $\bar{v}_{isQSSA}$ versus $\frac{1}{t_r} ln\left(\frac{s_0-p}{s_0}\right)$ (type I) or $\frac{1}{t_r} ln\left(\frac{s}{s_0}\right)$ (type II) to obtain $K_M$ and $v_{max}$. In the later sections, we will show that type II form gives more accurate estimate of $K_M$ than type I. The conditions for the validity of **Eq. 11** are $(\eta, \varepsilon) \to (0,0)$ and $t_c < t_r < t_m$. One can also integrate $v_{sQSSA} \cong -\frac{ds}{dt} \cong \frac{v_{max}s}{K_M+s}$ given in **Eqs. 5** under the conditions $\varepsilon \to 0$ for the initial condition s = $s_0$ at t = 0 and derive the following expression for the average reaction velocity.

$$\bar{v}_{csQSSA} \cong \frac{p_{csQSSA}}{t_r} = \frac{1}{t_r}\left(s_0 - s_{csQSSA} - \frac{v_{csQSSA}}{k_2}\right); s_{csQSSA} \cong K_M W\left(\frac{s_0}{K_M} exp\left(\frac{s_0 - v_{max}t_r}{K_M}\right)\right). \quad [12]$$

In this equation, we have used the mass conservation law $p = (s_0 - s - v/k_2)$ and $v_{csQSSA} = -\frac{ds_{csQSSA}}{dt} = \frac{v_{max}s_{csQSSA}}{K_M+s_{csQSSA}}$. When $\left(\frac{e_0}{K_M+s_{csQSSA}}\right) \ll 1$, then one finds from **Eqs. 12** that $\bar{v}_{csQSSA} \cong \frac{1}{t_r}\left(s_0 - s_{csQSSA}\right)$ where $p_{csQSSA} \cong \left(s_0 - s_{csQSSA}\right)$ which is an alternate form on **Eq. 11**. In **Eqs. 12**, W(x) is the Lambert W function [33, 34] which is the solution of y exp(y) = x for y. The conditions of validity of **Eqs. 12** is similar to **Eqs. 11**. We use **Eq. 11** for comparison purposes since **Eq. 11** is simple and computationally cheaper than **Eq. 12**. $K_M$ and $v_{max}$ can be obtained from the steady state datasets using any one of the expressions $(\bar{v}_{sQSSA}, \bar{v}_{isQSSA}, \bar{v}_{\varepsilon}, \bar{v}_{csQSSA})$ under appropriate validity conditions and timescales. However, except the condition on $\varepsilon$, all the others conditions (on $\eta$, $t_c$, $t_m$) require *a priori* knowledge on $k_1$, $k_2$ and $K_M$ which are generally not available for unknown enzyme systems. In this scenario, one can perform direct non-linear least square fitting of the time course kinetic data on (p, t) or (s, t) with the system of nonlinear ODEs given in **Eqs. 1** to obtain $k_1$, $k_{-1}$ and $k_2$. However, accuracy of such fitting procedure strongly depends on the detailed data points on both pre- and post-steady state regimes that warrants progress curve data collection on sub-millisecond timescales apart from the steady state timepoint.

## Numerical Simulation

The stochastic version of nonlinear ODEs given in **Eqs. 1.1-1.4** can be numerically integrated using the following Euler type scheme on chemical Langevin equations [35, 36].

$$x_{n+1} = x_n + (k_1(e_0 - x_n)(s_0 - x_n - p_n) - (k_{-1} + k_2)x_n)\Delta t + \Gamma_{x,t}. \quad [13.1]$$

$$\Gamma_{x,t} = \rho\xi_{p,t}\sqrt{(k_1(e_0 - x_n)(s_0 - x_n - p_n) - (k_{-1} + k_2)x_n)\Delta t}. \quad [13.2]$$

$$p_{n+1} = p_n + k_2 x_n \Delta t + \rho\sqrt{k_2 x_n \Delta t}\xi_{p,t}. \quad [13.3]$$

$$e_n = (e_0 - x_n); s_n = (s_0 - x_n - p_n);\ \langle \xi_{p,t} \rangle = 0;\ \langle \xi_{x,t} \rangle = 0\ . \qquad [13.4]$$
$$\langle \xi_{p,t} \xi_{p,t'} \rangle = \delta(t - t');\ \langle \xi_{x,t} \xi_{x,t'} \rangle = \delta(t - t');\ \langle \xi_{x,t} \xi_{p,t'} \rangle = 0. \qquad [13.5]$$

Here the initial conditions are $x = 0$, $s = s_0$, $e = e_0$, and $p = 0$ at $t = 0$, $\xi_{q,t}$ where q = (x, p) are delta correlated gaussian white noise terms and $\rho$ is the additional noise control parameter. The time increment parameter $\Delta t$ will be adjusted to capture both pre- as well as post-steady state dynamics. Setting $\rho = 0$, results in the deterministic trajectory of the MMS scheme for the given set of parameters and initial conditions. To check the consistency among the data fitting methods ($\bar{v}_{sQSSA}, \bar{v}_{isQSSA}, \bar{v}_{\varepsilon}$), several stochastic trajectories of **Eqs. 1** were generated with fixed $t_r$ and $e_0$ and different $s_0$ values. Stochastic simulated datasets are posted at (DOI: 10.5281/zenodo.15094623) and the sample code implementations of ML algorithm are given in the **Supporting Materials**. To obtain the kinetic parameters $k_1$, $k_{-2}$ and $k_2$ from the progress curve data with initial $s_0$ and $e_0$, we used nonlinear least square fitting [37] using Marquardt-Levenberg (ML) algorithm [38, 39]. The computational workflow of ML scheme starts with minimizing the function $L = \sum_{i=1}^{n} (f(t_i|r) - y_i)^2$ where $f(t|r)$ is the numerical solution to **Eqs. 1** with the parameter set $r = [k_1, k_{-1}, k_2]$, and $y$ is the observed dataset where $(f, y) = (p, s)$. Upon expanding $L$ in Taylor series around $r_u = [k_1^u, k_{-1}^u, k_2^u]$ and then truncating with the first order terms, one finds the following approximation.

$$L \cong \sum_{i=1}^{n} \left( \Delta y_i + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_1^u - k_1) + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_{-1}^u - k_{-1}) + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_2^u - k_2) \right)^2 . \quad [14]$$

In this equation $\Delta y_i = f(t_i|r_u) - y_i$. The conditions for the minimum with respect to $r$ will be $\frac{\partial L}{\partial k_1} = 0$; $\frac{\partial L}{\partial k_{-1}} = 0$; $\frac{\partial L}{\partial k_2} = 0$ which can be explicitly written as follows.

$$\sum_{i=1}^{n} \left( \Delta y_i + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_1^u - k_1) + \frac{\partial f(t_i|r_u)}{\partial k_{-1}}(k_{-1}^u - k_{-1}) + \frac{\partial f(t_i|r_u)}{\partial k_2}(k_2^u - k_2) \right) \frac{\partial f(t_i|r_u)}{\partial k_1} = 0\ . \quad [15.1]$$
$$\sum_{i=1}^{n} \left( \Delta y_i + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_1^u - k_1) + \frac{\partial f(t_i|r_u)}{\partial k_{-1}}(k_{-1}^u - k_{-1}) + \frac{\partial f(t_i|r_u)}{\partial k_2}(k_2^u - k_2) \right) \frac{\partial f(t_i|r_u)}{\partial k_{-1}} = 0. \quad [15.2]$$
$$\sum_{i=1}^{n} \left( \Delta y_i + \frac{\partial f(t_i|r_u)}{\partial k_1}(k_1^u - k_1) + \frac{\partial f(t_i|r_u)}{\partial k_{-1}}(k_{-1}^u - k_{-1}) + \frac{\partial f(t_i|r_u)}{\partial k_2}(k_2^u - k_2) \right) \frac{\partial f(t_i|r_u)}{\partial k_2} = 0\ . \quad [15.3]$$

**Eqs. 15.1-3** can be written in a matrix form [37] as $J^T \Delta y + (J^T J)\Delta r = 0$. Definition of various terms in this matrix equation are given below.

$$J = \begin{bmatrix} \frac{\partial f(t_1|r_u)}{\partial k_1} & \frac{\partial f(t_1|r_u)}{\partial k_{-1}} & \frac{\partial f(t_1|r_u)}{\partial k_2} \\ \vdots & \vdots & \vdots \\ \frac{\partial f(t_n|r_u)}{\partial k_1} & \frac{\partial f(t_n|r_u)}{\partial k_{-1}} & \frac{\partial f(t_n|r_u)}{\partial k_2} \end{bmatrix};\ \Delta r = \begin{bmatrix} k_1^u - k_1 \\ k_{-1}^u - k_{-1} \\ k_2^u - k_2 \end{bmatrix};\ \Delta y = \begin{bmatrix} f(t_1|r_u) - y_1 \\ \vdots \\ f(t_n|r_u) - y_n \end{bmatrix}. \qquad [16]$$

We compute the partial derivatives with respect to parameters $[k_1, k_{-1}, k_2]$ at time point $t_i$ as follows where $f(t|r)$ is the numerical integrated solution to **Eqs. 1** with the parameter set $r = [k_1, k_{-1}, k_2]$.

$$\frac{\partial f(t_i|r_u)}{\partial k_1} = \frac{\partial f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\partial k_1} \cong \frac{f(t_i|[k_1^u+\alpha,k_{-1}^u,k_2^u])-f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\alpha} \ . \quad [17.1]$$

$$\frac{\partial f(t_i|r_u)}{\partial k_{-1}} = \frac{\partial f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\partial k_{-1}} \cong \frac{f(t_i|[k_1^u,k_{-1}^u+\alpha,k_2^u])-f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\alpha} \ . \quad [17.2]$$

$$\frac{\partial f(t_i|r_u)}{\partial k_2} = \frac{\partial f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\partial k_2} \cong \frac{f(t_i|[k_1^u,k_{-1}^u,k_2^u+\alpha])-f(t_i|[k_1^u,k_{-1}^u,k_2^u])}{\alpha} \ . \quad [17.3]$$

We set $\alpha$ to the lowest possible value depending on the computational accuracy requirement. In **Eqs. 16**, *J* is the Jacobian matrix and explicitly one finds the following.

$$J^T\Delta y = \begin{bmatrix} \sum_{i=1}^n \left(\Delta y_i \frac{\partial f(t_i|r_u)}{\partial k_1}\right) \\ \sum_{i=1}^n \left(\Delta y_i \frac{\partial f(t_i|r_u)}{\partial k_{-1}}\right) \\ \sum_{i=1}^n \left(\Delta y_i \frac{\partial f(t_i|r_u)}{\partial k_2}\right) \end{bmatrix} . \quad [18]$$

$$J^TJ = \begin{bmatrix} \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_1}\frac{\partial f(t_i|r_u)}{\partial k_1}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\frac{\partial f(t_i|r_u)}{\partial k_1}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_2}\frac{\partial f(t_i|r_u)}{\partial k_1}\right) \\ \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_1}\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_2}\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\right) \\ \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_1}\frac{\partial f(t_i|r_u)}{\partial k_2}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_{-1}}\frac{\partial f(t_i|r_u)}{\partial k_2}\right) & \sum_{i=1}^n \left(\frac{\partial f(t_i|r_u)}{\partial k_2}\frac{\partial f(t_i|r_u)}{\partial k_2}\right) \end{bmatrix} . \quad [19]$$

From **Eqs. 15-19**, one obtains the following Marquardt-Levenberg (ML) iterative scheme where $\lambda$ is the ML parameter and $\lambda = 0$ represents the Newton's method [37].

$$r_{u+1} = r_u + \left(J^TJ + \lambda\ \mathrm{diag}(J^TJ)\right)^{-1} J^T\Delta y \ . \quad [20]$$

The reduced sum of squares of deviation will be computed at the end of u$^{th}$ iteration as $Z_u = \sum_{i=1}^n \left(y_{u,i} - y_i\right)^2$ where $y_{u,i} = f(t_i|r_u)$ is the predicted data point values for $r_u$ which will be obtained by numerical integration of **Eqs. 1** using Euler type scheme given in **Eqs. 13** with ρ = 0. Here λ should be adjusted depending on changing pattern of $Z_u$ over iterations. When $Z_u < Z_{u+1}$, then we set $\lambda \to \lambda\delta$. When $Z_u > Z_{u+1}$ then we set $\lambda \to \lambda/\delta$. Here the $\delta$ is the ML tuning parameter [38]. When $|Z_u - Z_{u+1}| < \theta$ where $\theta$ is the required tolerance limit, then we exit from iteration and the final parameters $\hat{r}$ will be obtained. For NLS fit of various datasets we used $\lambda = 10^{-2}$, $\delta = 2$ and $\theta = 10^{-6}$. the final sum of squares error (SSE) and the covariance matrix will be calculated as follows.

$$SSE = \sum_{i=1}^n (\hat{y}_i - y_i)^2 \ ; \ \hat{\sigma}^2 = \frac{SSE}{n-3}; \ Cov(\hat{r}) = (J^TJ)^{-1}\hat{\sigma}^2. \quad [21]$$

Upon computing the total sum of squares (SST), one can compute the final nonlinear regression coefficient $R^2$ as follows.

$$SST = \sum_{i=1}^n y_i^2 - \frac{1}{n}\left(\sum_{i=1}^n y_i\right)^2; \ R^2 = 1 - \frac{SSE}{SST} \quad [22]$$

Here $\hat{y}_i = f(t_i|\hat{r})$ where $\hat{r} = \left[\hat{k}_1, \hat{k}_{-1}, \hat{k}_2\right]$ is the final fit parameter values. We use the final SSE and $R^2$ to compare the goodness of fit across trajectories with different $s_0$ and replications.

Similar workflow was used to obtain $K_M$ and $v_{max}$ using $\bar{v}_{\varepsilon}$ as given in **Eq. 10** with the median of $k_1$ obtained from the progress curve analysis across different $s_0$ over several replications outlined in **Eqs. 14-21**. Linear least square fitting with **Eq. 11** [37] for $\bar{v}_{isQSSA}$ yields the required parameter estimates along with the errors.

## Results and Discussion

There are several issues and limitations associated with the currently available steady state as well as progress curve methods to estimate the $K_M$ and $v_{max}$ of single substrate MM enzymes. First let us consider sQSSA with stationary reactant assumption ($\bar{v}_{sQSSA}$) which is valid only when $t_c \ll t_r < t_m$ and $(\eta, \varepsilon) \to (0,0)$. Since $K_M$ values are not generally available for the unknown enzyme systems, to achieve these criteria, one needs to set higher values for $s_0$ than $e_0$ which will lead to high background fluctuations in the measured average reaction velocities. For example, let $\bar{v}_0 = (\bar{v}_{0,1}, \bar{v}_{0,2})$ correspond to respectively $s_0 = (s_{0,1}, s_{0,2})$. When $s_0 \to \infty$, then one finds the following.

$$lim_{s_0 \to \infty} \left| \bar{v}_{0,2} - \bar{v}_{0,1} \right| = lim_{s_0 \to \infty} v_{max} \left| \frac{s_{0,2}}{K_M + s_{0,2}} - \frac{s_{0,1}}{K_M + s_{0,1}} \right| \to 0. \qquad [23]$$

**Eq. 23** leads to high noise level in the observed average reaction velocities that in turn results in high level of uncertainty in the estimation of $K_M$. Similar effects are prevalent in the progress curve analysis using especially the trajectories on ($p$, $t$) space since $p \cong 0$ in the pre-steady state regime that leads to high level of background fluctuations in the measured values of $p$. This in turn results in high uncertainty in the estimation of $K_M$ using the data on ($p$, $t$) which significantly depends on the pre-steady state dynamics. Clearly, the data on ($p$, $t$) can be used only in the steady state methods which mainly work well in the post-steady state regime. Detailed NLS fitting analysis of the data on ($p$, $t$) and ($s$, $t$) spaces in the presence as well as absence of noise suggests the following.

a) Progress curve methods using Marquardt-Levenberg algorithm outlined in the methods section works very well over data on (s, t) as well as (p, t) in the absence of noise i.e. $\rho = 0$ in **Eqs. 13** as demonstrated in **Figs. 2A** and **2B**. NLS fitting over the progress curve data on (p, t) as well as (s, t) can recover the original $K_M$ and $v_{max}$ with minimal error.
b) When $\rho > 0$, then the NLS fit procedure work better with the data on (s, t) than the data on (p, t) as demonstrated in **Figs. 2C-F**. Particularly, fluctuations in the trajectories of (p, t) in the pre-steady state regime is demonstrated in **Fig. 2C**. NLS fitting methods can capture $k_1$ and $k_2$ very well and the estimate on $k_{-1}$ is prone to severe uncertainties especially when $k_{-1} \ll (k_1, k_2)$ as demonstrated with sample trajectories from **Figs. 2C** and **2D** in **Figs. 2E** and **2F**. This in turn leads to significant amount error in the estimation of $K_M$ although $v_{max}$ is not much affected by this issue.
c) The distribution of NLS fit parameters obtained at different noise levels are shown in **Figs. 3** for both (s, t) and (p, t) time course datasets. These results clearly suggest that $K_M$ values obtained from the NLS fit of time course dataset on (p, t) space are unreliable and prone to significant fluctuations in the presence of high noise levels. When the noise level increases, then the distribution of fit parameters corresponding to (p, t) space dataset exhibits bimodality as shown in (**Figs. 3D**, **3F**). Whereas, NLS fit over time course

dataset on (s, t) space can yield reliable median estimates of $K_M$ even at high noise levels. Particularly, monomodal type distribution of fit values of $K_M$ and $v_{max}$ are observed for the dataset on (s, t) space as demonstrated in **Figs. 3A**, **3C**, **3E**.

d) Remarkably both time course datasets on (s, t) and (p, t) spaces can yield reliable NLS fit median estimates of $v_{max}$ across various noise levels as demonstrated in **Fig. 3**. This observation is reasonable since $v_{max}$ mostly relies on the post-steady state dynamics and therefore the fluctuations in the pre steady state regime of (p, t) space will not influence the error in the estimation of $v_{max}$.

These observations suggest that neither progress curve analysis alone nor steady state methods alone can provide consistent estimates of $K_M$. Multiple ways of analyzing the same dataset of (p, t) and (s, t) is clearly required to check for the consistency across various steady state and progress curve methods and obtain reliable estimates. In this context, one can use the median value of $k_1$ obtained from progress curve methods into the NLS fit on $\bar{v}_\varepsilon$ as given in **Eq. 10** to obtain consistent estimates of $K_M$ and $v_{max}$ across progress curve analysis and steady state methods. We consider the following methods I, II, III and IV using the same main time course datasets on (s, p, t). For Methods II, III and IV, sub-dataset on average reaction velocity versus $s_0$ at different $t_r$ will be generated from the main dataset. Here the average velocity is defined as (total product formed at the end of $t_r$) / $t_r$ where $t_r$ will be iterated approximately from the steady state timescale $t_c$ to $t_m$. Various NLS fit methods I, II, III and IV are summarized as follows.

a) **Method I** is the direct NLS fitting of the main progress curve dataset on (s, t) with fixed $e_0$ and different $s_0$ across several replications with the original rate equations **Eqs. 1.1-1.4** using Marquardt-Levenberg algorithm as outlined in the methods section. $K_M$ values will be calculated from the fit values of ($k_1$, $k_{-1}$, $k_2$) and the median of ($k_1$, $k_{-1}$, $k_2$, $K_M$) values will also be obtained.
b) In **Method II**, the parameter $k_1$ in the definition of $\bar{v}_\varepsilon$ of **Eqs. 10** will be fixed with the median of $k_1$ values obtained from Method I and the sub-dataset on ($\bar{v}$, $s_0$) computed from the main time course dataset on (p, t) at different $t_r$ will be used for NLS fitting with fixed $k_1$ to obtain $K_M$ and $v_{max}$.
c) In **Method III**, $\bar{v}_{isQSSA}$ as defined in **Eq. 11** will be used to fit the sub-dataset on ($\bar{v}$, $p$, $s_0$) computed from the main dataset on (p, t) at different $t_r$ and $s_0$. One can also use $\bar{v}_{csQSSA}$ that is computationally costlier than $\bar{v}_{isQSSA}$.
d) In **Method IV** that is widely used across various literature, $\bar{v}_{sQSSA} \cong v_0$ will be used to fit the sub-dataset on ($\bar{v}$, $s_0$) computed from the main dataset on (p, t) at different reaction time $t_r$ and $s_0$.

In Methods II, III and IV, the $K_M$ value corresponding to $t_r$ with minimum sum of square error (SSE) as given in **Eq. 21** will be computed as demonstrated in **Fig. 4**. There exists a critical reaction time $t_r$ at which the sum of square error in the estimation of $K_M$ attains the minimum which is clearly observed in the absence of noise in the dataset ($\rho = 0$, **Fig. 4A**). Remarkably, **Fig. 4B** suggest that the error in the estimation of $K_M$ strongly depends on the reaction time $t_r$ in steady state Methods II, III and IV. In the presence of noise in the dataset, the median of SSE corresponding to $K_M$ decreases monotonically and the fit parameter value converges to

the original $K_M$ as $t_r$ is increased towards $t_m$. The speed of convergence of fit $K_M$ towards the original $K_M$ is much faster in Method II than in Methods III and IV as demonstrated in **Fig. 4B**. Sum of square errors of various approximations can be defined as $SSE = \sum_{i=0}^{n}\left(\bar{v}_{t_i} - \bar{v}_{G,t_i}\right)^2$ where $t_n = t_r$ and $\bar{v}_{t_i}$ is the numerically simulated average velocity at time t = t, and $\bar{v}_{G,t_i}$ are the approximations where the subscripts G = sQSSA, csQSSA, ε, isQSSA (I, II) associated with $\bar{v}_{sQSSA}, \bar{v}_{csQSSA}, \bar{v}_{\varepsilon}, \bar{v}_{isQSSA}$. The computed $SSE$ seems to be strongly dependent on $t_r$ and range of iteration of $s_0$ as demonstrated in **Fig. 4C** and **4D**. Clearly, SSE of $\bar{v}_{\varepsilon}$ is minimum near $t_c$ and the SSEs corresponding to $(\bar{v}_{isQSSA}\ (I, II), \bar{v}_{csQSSA})$ attain minimum approximately near $t_m$. The timescale $t_Q$ at which the SSE of $\bar{v}_{sQSSA}$ attains minimum can be computed by solving $\bar{v}_{\varepsilon} - \bar{v}_{csQSSA} = 0$ for t as shown in **Fig. 4E**. This is the critical time point at which both pre- and post-steady state approximations $(\bar{v}_{\varepsilon}, \bar{v}_{csQSSA})$ start to deviate from the original simulated average reaction velocity values.

When $t_r = t_Q$, then the reaction velocity will be $\bar{v} \cong v_0$ leading to minimum amount of error in the approximation $\bar{v}_{sQSSA}$. Remarkably, all the fitting methods $(\bar{v}_{sQSSA}, \bar{v}_{isQSSA}, \bar{v}_{\varepsilon}, \bar{v}_{csQSSA})$ approximately perform well near $t_Q$, at which the variation of the fit parameters across these methods will be at minimum. The entire procedure will be repeated with different $e_0$ until obtaining consistent median estimates of $K_M$ and $v_{max}$ with minimal threshold variation across methods I to IV. We consider the coefficient of variation (CV = standard deviation / mean) of the median $K_M$ values obtained from methods I-IV as the *reliability index*. **Figs. 4C**, **4D** and **4E** suggest that the reliability index will be at minimum when the reaction time is set as $t_r = t_Q$. In the absence of *a priori* knowledge on enzyme kinetic parameters, one can use the reliability index to obtain best fit values of $K_M$ which occurs near the optimum reaction time $t_Q$. Upon obtaining such consistent median estimates of $K_M$ corresponding to a given threshold *reliability index*, all the $K_M$ values obtained from the same dataset using different methods will be then pooled and the overall median of $K_M$ and $v_{max}$ will be computed as depicted in **Fig. 5**.

The distributions of various fit parameters obtained from Methods I to IV over several replications are shown in **Fig. 6** along with the overall pooled distribution of $K_M$ and $v_{max}$. The distribution of fit parameters ($k_1$, $k_{-1}$, $k_2$) show skewness towards lower values (**Figs. 6A1-3**). Particularly, the distribution of $k_{-1}$ exhibits a bimodal type with zero spike. As result, the distribution of $K_M$ also exhibits spike near zero and $v_{max}$ exhibits skewed distribution (**Figs. 6B1-2**). Since the median of $k_1$ obtained from Method I is used for the NLS fitting calculations in Method II, the distribution of $K_M$ values obtained from Method II also exhibits a spike near zero as shown in **Fig. 6B-3**. Interestingly, both $K_M$ and $v_{max}$ obtained from Method III exhibit symmetric type distribution (**Figs. 6C-1**, **6D-1**). However, the distribution of $K_M$ values obtained using Method III span towards negative values (**Fig. 6C-1**). The distribution of $K_M$ values obtained from Method IV also exhibit a spike near zero (**Fig. 6C-2**). Clearly, the inconsistency among the $K_M$ values obtained from different methods originates mainly from the type of distribution of fit parameter in the presence of noise.

Efficient progress curve fitting also seems to be dependent on the detailed data collection around the transition regions in (s, t) and (p, t) spaces with maximum change in curvature [31]. Clearly, there exists at least three different timescale regimes in the trajectory on (p, t) space with significant change in curvature viz. exponential increasing phase in the pre-steady state regime, linear increasing phase in the steady state region and reaction-ending phase.

Similar timescale regimes also exist in (s, t) space. Significant change in the curvature occurs at the interface of these timescale regimes approximately near $t_c$ and $t_m$. Accurate values of these interface timescales can be calculated from $s_\varepsilon$ and $s_{csQSSA}$ for the pre- and post-steady state regimes respectively. The approximate curvature $\zeta_t$ associated with $s_\varepsilon$ of **Eq. 8** can be defined as follows [40].

$$\zeta_t \cong \left(\frac{d^2 s_\varepsilon}{dt^2}\right) \Big/ \left(1 + \left(\frac{ds_\varepsilon}{dt}\right)^2\right)^{3/2} ;\ s_\varepsilon \cong s_0 - p_\varepsilon - v_\varepsilon / k_2 \qquad [24]$$

Here $p_\varepsilon$ and $v_\varepsilon$ are defined as in **Eqs. 8**. Upon solving $\frac{d\zeta_t}{dt} = 0$ for t, one finds the time scale associated with the maximum change in curvature at the interface of pre-steady to steady steady-state transition $t_R$ as follows.

$$t_R \cong -t_c ln\left(\frac{t_c k_2}{4v_0(t_c k_2 - 1)}\left(v_0 - \sqrt{9v_0^2 + 8}\right)\right);\ v_0 = \frac{v_{max} s_0}{K_M + s_0} \qquad [25]$$

Similarly noting the definition of $s_{csQSSA}$ as given in **Eqs. 12**, the approximate curvature $\psi_t$ in the steady state to post-steady state transition regime of (s, t) space can be defined as follows.

$$\psi_t \cong \left(\frac{d^2 s_{csQSSA}}{dt^2}\right) \Big/ \left(1 + \left(\frac{ds_{csQSSA}}{dt}\right)^2\right)^{3/2} \qquad [26]$$

Upon solving $\frac{d\psi_t}{dt} = 0$ for t, one finds the timescale associated with the maximum change in curvature at the interface of steady state to reaction ending regime $t_U$ as follows.

$$t_U \cong \frac{1}{v_{max}}\left| s_0 - K_M\left(h + ln\left(\frac{K_M h}{s_0}\right)\right)\right|;\ h = \left(\sqrt{8v_{max}^2 + 9} - 1\right)/4(v_{max}^2 + 1) \qquad [27]$$

Since the enzyme system evolves with constant velocity $v_0$ in the timescale region $t_R < t < t_U$, progress curve fitting methods can yield reliable estimate of $K_M$ only when the experimental data collection is done around the timescales ($t_R$, $t_U$) with maximum change in the curvature of (s, t) trajectory. Whereas, steady state methods work very well in the constant velocity regime with minimal change in the curvature of the trajectory of (s, t). Change in curvature of the trajectory in (s, t) space with respect to time is shown in **Fig. 7A** along with the approximations given in **Eqs. 25-27** which are in good agreement with the numerical simulation results. Further, **Eqs. 25** and **27** suggest that ($t_R$, $t_U$) will be close to ($t_c$, $t_m$) and for practical purposes we use ($t_c$, $t_m$) for checking the validity conditions of various model fitting methods. The simulated average reaction velocity (p / t) along with the approximations ($\bar{v}_{sQSSA}, \bar{v}_\varepsilon, \bar{v}_{isQSSA}\ or\ \bar{v}_{csQSSA}$) are shown in **Fig. 7B**. These suggest that the approximation $\bar{v}_\varepsilon$ works well in the pre-steady to steady state regime and $\bar{v}_{isQSSA}\ or\ \bar{v}_{csQSSA}$ work well in the post steady state regime and $\bar{v}_{sQSSA}$ work only in narrow range of timescales with constant reaction velocity. Clearly, the reaction time $t_r$ plays important role in deciding the reliability index among Methods I to IV. The error in the estimation of $K_M$ using $\bar{v}_\varepsilon$ proportionally increases when $t_r$ increases well beyond the steady state. Conversely, inclusion of data points from the pre-steady state regime increases the error in $\bar{v}_{isQSSA}\ or\ \bar{v}_{csQSSA}$. These means that

the error in the estimation of KM using the widely used $\bar{v}_{sQSSA}$ will be always higher than the error associated with $\left(\bar{v}_{\varepsilon}, \bar{v}_{isQSSA}\ or\ \bar{v}_{csQSSA}\right)$. Reliability index can be used as a check point to optimize the appropriate $t_r$ to obtain accurate estimate of $K_M$ values from progress curve and various steady state methods.

Software packages such as global kinetic explorer and dynafit [41-43] were developed to fit the data obtained from progress curve and steady state experiments. Several comparative studies were done earlier over steady state and progress curve methods to optimize the analysis workflow towards minimizing the estimation error [44, 45]. Selective removal of data points [46] over certain progress curve regime were also proposed. However, all these methods are solely either progress curve or steady state type and there is no way to check the consistency of fit parameters across various methods since each method requires different type of input dataset. Accurate estimation of kinetic parameters depends on strict impose of validity conditions of each method as given in **Table 1**. Unfortunately, these validity conditions require *a priori* knowledge on various enzyme kinetic parameters which are not generally available for unknown systems. Since the conditions of validity for each method is different, a global analysis workflow with a common input dataset is required to obtain consistent estimates. In this context, our proposed computational workflow depicted in **Fig. 5** can yield reliable estimates of $K_M$ from the common time course dataset at different initial substrate concentrations $s_0$ and replications. The reliability index which is the coefficient of variation of median $K_M$ values obtained from Methods I-IV can act as a check point against unreliable estimates of $K_M$. When the computed reliability index is more than the given threshold value, then one needs to iterate the reaction time until obtaining the desired threshold level. Clearly, the proposed methods do not require *a priori* knowledge on $K_M$ values unlike the currently available sQSSA methods.

## Conclusion

Accurate estimation of enzyme kinetic parameters such as $K_M$ and $v_{max}$ from the experimental datasets is critical to characterize any enzyme. The validity of the widely used standard quasi steady state method over decades relies on the *a priori* information of $K_M$ and other kinetic parameters which are generally not available for newly identified enzyme systems. Progress curve methods were proposed to alleviate such issues. However, nonlinear least square fitting of the time course dataset with the original differential rate equations works well only in the timescale regimes with maximum change in the curvature of the trajectory of substrate evolution. Identifying these timescales requires the knowledge on enzyme kinetic parameters. Though experimental trial and error methods can be used to identify such timescales, they are generally cost ineffective.

Further, progress curve methods work better with the time course data on substrate than the time course data on product since the product concentration is close to zero in the pre-steady state regime. Since steady state and progress curve analysis use different types of datasets in the current analysis set up, the estimated $K_M$ values will be inconsistent among these methods. This leads to the dilemma of identifying correct $K_M$ values among various steady state and progress curve analyses. Here we have shown that the error in the estimation of $K_M$ using steady state methods is strongly dependent on the total reaction time. We have further shown that the steady state methods work well over the reaction timescales with minimum change in the curvature of the trajectory of product evolution with constant reaction velocity.

Considering these theoretical factors, we have developed an integrated method of analysis which comprises of both progress curve and steady state analysis using the same time course datasets that also consider the total reaction time. We considered three different methods of steady state analyses that involve fitting of the average reaction velocity with the total substrate concertation. We have defined the reliability index which is the coefficient of variation of the median $K_M$ values obtained from progress curve and other steady state fitting methods using the same time course dataset at different substrate concentrations and replications. We further demonstrate that reliable estimate of $K_M$ can be obtained by iterating the total reaction time to minimize the reliability index across time course and steady state curve fitting methods.

## References

1. Alberts B. Molecular biology of the cell. New York: Garland Science; 2002.
2. Stryer L. Biochemistry. New York: W.H. Freeman; 1988.
3. Voet D, Voet JG. Biochemistry. New York: J. Wiley & Sons; 1995.
4. Michaelis L, Menten ML. {Die kinetik der invertinwirkung}. Biochem Z. 1913;49(333-369):352. doi: citeulike-article-id:5936552.
5. Briggs GE, Haldane JBS. A Note on the Kinetics of Enzyme Action. Biochemical Journal. 1925;19(2):338-9. PubMed PMID: PMC1259181.
6. Murugan R. Solution to Michaelis–Menten enzyme kinetic equation via undetermined gauge functions: Resolving the nonlinearity of Lineweaver–Burk plot. The Journal of chemical physics. 2002;117(9):4178-83.
7. Murray JD. Mathematical biology. New York: Springer; 2002.
8. Fraser SJ. Slow manifold for a bimolecular association mechanism. J Chem Phys. 2004;120(7):3075-85. doi: 10.1063/1.1634555. PubMed PMID: 15268460.
9. Roussel MR, Fraser SJ. Invariant manifold methods for metabolic model reduction. Chaos. 2001;11(1):196-206. doi: 10.1063/1.1349891. PubMed PMID: 12779453.
10. Dingee JW, Anton AB. A new perturbation solution to the Michaelis-Menten problem. AIChE Journal. 2008;54(5):1344-57. doi: 10.1002/aic.11461.
11. Vogt D. A simplified perturbation solution of Michaelis-Menten kinetics equations in a “total” framework. Journal of Mathematical Chemistry. 2013;51(9):2257-63. doi: 10.1007/s10910-013-0221-0.
12. Dell’Acqua G, Bersani AM. A perturbation solution of Michaelis–Menten kinetics in a “total” framework. Journal of Mathematical Chemistry. 2012;50(5):1136-48. doi: 10.1007/s10910-011-9957-6.
13. Seshadri MS, Fritzsch G. The time evolution of sequential enzyme reactions: a singular perturbation approach. J Theor Biol. 1981;93(1):197-205. PubMed PMID: 7334820.
14. Segel LA, Slemrod M. The Quasi-Steady-State Assumption: A Case Study in Perturbation. SIAM Review. 1989;31(3):446-77. doi: 10.1137/1031091.
15. Dell’Acqua G, Bersani AM. On the appropriate use of asymptotic expansions in enzyme kinetics. Journal of Mathematical Chemistry. 2014;52(10):2475-81. doi: 10.1007/s10910-014-0403-4.
16. Murugan R. Theory on the rate equation of Michaelis–Menten type single-substrate enzyme catalyzed reactions. Journal of Mathematical Chemistry. 2018;56(2):508-56. doi: 10.1007/s10910-017-0791-3.
17. Hofstee BH. Non-inverted versus inverted plots in enzyme kinetics. Nature. 1959;184:1296-8. doi: 10.1038/1841296b0. PubMed PMID: 14402470.
18. Atkins GL, Nimmo IA. A comparison of seven methods for fitting the Michaelis-Menten equation. Biochem J. 1975;149(3):775-7. PubMed PMID: 1201002; PubMed Central PMCID: PMCPMC1165686.
19. Lineweaver H, Burk D. The Determination of Enzyme Dissociation Constants. Journal of the American Chemical Society. 1934;56(3):658-66. doi: 10.1021/ja01318a036.
20. Wilkinson GN. Statistical estimations in enzyme kinetics. Biochem J. 1961;80(2):324-32. doi: 10.1042/bj0800324. PubMed PMID: 13785321; PubMed Central PMCID: PMCPMC1244002.
21. Fernley HN. Statistical estimations in enzyme kinetics. The integrated Michaelis equation. Eur J Biochem. 1974;43(2):377-8. doi: 10.1111/j.1432-1033.1974.tb03423.x. PubMed PMID: 4838986.

22. Murugan R. Theory on the rate equations of Michaelis-Menten type enzyme kinetics with competitive inhibition. PLOS ONE. 2024;19(7):e0302679. doi: 10.1371/journal.pone.0302679.
23. Segel LA. On the validity of the steady state assumption of enzyme kinetics. Bull Math Biol. 1988;50(6):579-93. PubMed PMID: 3219446.
24. Borghans JM, De Boer RJ, Segel LA. Extending the quasi-steady state approximation by changing variables. Bulletin of Mathematical Biology. 1996;58(1):43-63. doi: http://dx.doi.org/10.1016/0092-8240(95)00306-1.
25. Grima R, Walter NG, Schnell S. Single-molecule enzymology a la Michaelis-Menten. FEBS J. 2014;281(2):518-30. doi: 10.1111/febs.12663. PubMed PMID: 24289171.
26. Choi IY, Lee SP, Kim SG, Gruetter R. In vivo measurements of brain glucose transport using the reversible Michaelis-Menten model and simultaneous measurements of cerebral blood flow changes during hypoglycemia. J Cereb Blood Flow Metab. 2001;21(6):653-63. doi: 10.1097/00004647-200106000-00003. PubMed PMID: 11488534.
27. Lee BY, Kwon KI, Kim MS, Baek IH. Michaelis-Menten elimination kinetics of etanercept, rheumatoid arthritis biologics, after intravenous and subcutaneous administration in rats. Eur J Drug Metab Pharmacokinet. 2016;41(4):433-9. doi: 10.1007/s13318-015-0270-9. PubMed PMID: 25725773.
28. Choi B, Rempala GA, Kim JK. Beyond the Michaelis-Menten equation: Accurate and efficient estimation of enzyme kinetic parameters. Scientific Reports. 2017;7(1):17018. doi: 10.1038/s41598-017-17072-z.
29. Frère JM, Verlaine O, Matagne A. The measurement of true initial rates is not always absolutely necessary to estimate enzyme kinetic parameters. Sci Rep-Uk. 2023;13(1). doi: ARTN 15053
10.1038/s41598-023-41805-y. PubMed PMID: WOS:001168651100052.
30. Duggleby RG, Clarke RB. Experimental designs for estimating the parameters of the Michaelis-Menten equation from progress curves of enzyme-catalyzed reactions. Biochim Biophys Acta. 1991;1080(3):231-6. PubMed PMID: 1954231.
31. Stroberg W, Schnell S. On the estimation errors of KM and V from time-course experiments using the Michaelis-Menten equation. Biophys Chem. 2016;219:17-27. doi: 10.1016/j.bpc.2016.09.004. PubMed PMID: 27677118.
32. Hanson SM, Schnell S. Reactant stationary approximation in enzyme kinetics. J Phys Chem A. 2008;112(37):8654-8. doi: 10.1021/jp8026226. PubMed PMID: 18714952.
33. Corless RM, Gonnet GH, Hare DEG, Jeffrey DJ, Knuth DE. On the LambertW function. Advances in Computational Mathematics. 1996;5(1):329-59. doi: 10.1007/BF02124750.
34. Abramowitz M, Stegun IA. Handbook of mathematical functions, with formulas, graphs, and mathematical tables. New York: Dover Publications; 1965.
35. Gardiner CW. Handbook of stochastic methods for physics, chemistry, and the natural sciences. Berlin; New York: Springer-Verlag; 1985.
36. Risken H. The Fokker-Planck equation : methods of solution and applications. Berlin; New York: Springer-Verlag; 1989.
37. Press WH. Numerical recipes in C++ : the art of scientific computing. 2nd ed. Cambridge, UK: Cambridge University Press; 2002.
38. Marquardt DW. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. Journal of the Society for Industrial and Applied Mathematics. 1963;11(2):431-41. doi: 10.1137/0111030.

39. Levenberg K. A method for the solution of certain non-linear problems in least squares. Quart Appl Math. 1944;2:164--8. doi: doi:10.1090/qam/10666.
40. Courant R, John F. Introduction to calculus and analysis. Repr. of the 1989 ed. Berlin: Springer; 1989.
41. Johnson KA. Fitting enzyme kinetic data with KinTek Global Kinetic Explorer. Methods Enzymol. 2009;467:601-26. doi: 10.1016/S0076-6879(09)67023-3. PubMed PMID: 19897109.
42. Johnson KA, Simpson ZB, Blom T. Global kinetic explorer: a new computer program for dynamic simulation and fitting of kinetic data. Anal Biochem. 2009;387(1):20-9. Epub 20081225. doi: 10.1016/j.ab.2008.12.024. PubMed PMID: 19154726.
43. Mak DA, Dunn S, Coombes D, Carere CR, Allison JR, Nock V, et al. Enzyme Kinetics Analysis: An online tool for analyzing enzyme initial rate data and teaching enzyme kinetics. Biochemistry and Molecular Biology Education. 2024;52(3):348-58. doi: https://doi.org/10.1002/bmb.21823.
44. Cho YS, Lim HS. Comparison of various estimation methods for the parameters of Michaelis-Menten equation based on in vitro elimination kinetic simulation data. Transl Clin Pharmacol. 2018;26(1):39-47. Epub 20180316. doi: 10.12793/tcp.2018.26.1.39. PubMed PMID: 32055546; PubMed Central PMCID: PMCPMC6989224.
45. Marasović M, Marasović T, Miloš M. Robust Nonlinear Regression in Enzyme Kinetic Parameters Estimation. Journal of Chemistry. 2017;2017(1):6560983. doi: https://doi.org/10.1155/2017/6560983.
46. Petric B, Golicnik M, Bavec A. The Removal of Time-Concentration Data Points from Progress Curves Improves the Determination of K(m): The Example of Paraoxonase 1. Molecules. 2022;27(4). Epub 20220215. doi: 10.3390/molecules27041306. PubMed PMID: 35209091; PubMed Central PMCID: PMCPMC8874660.

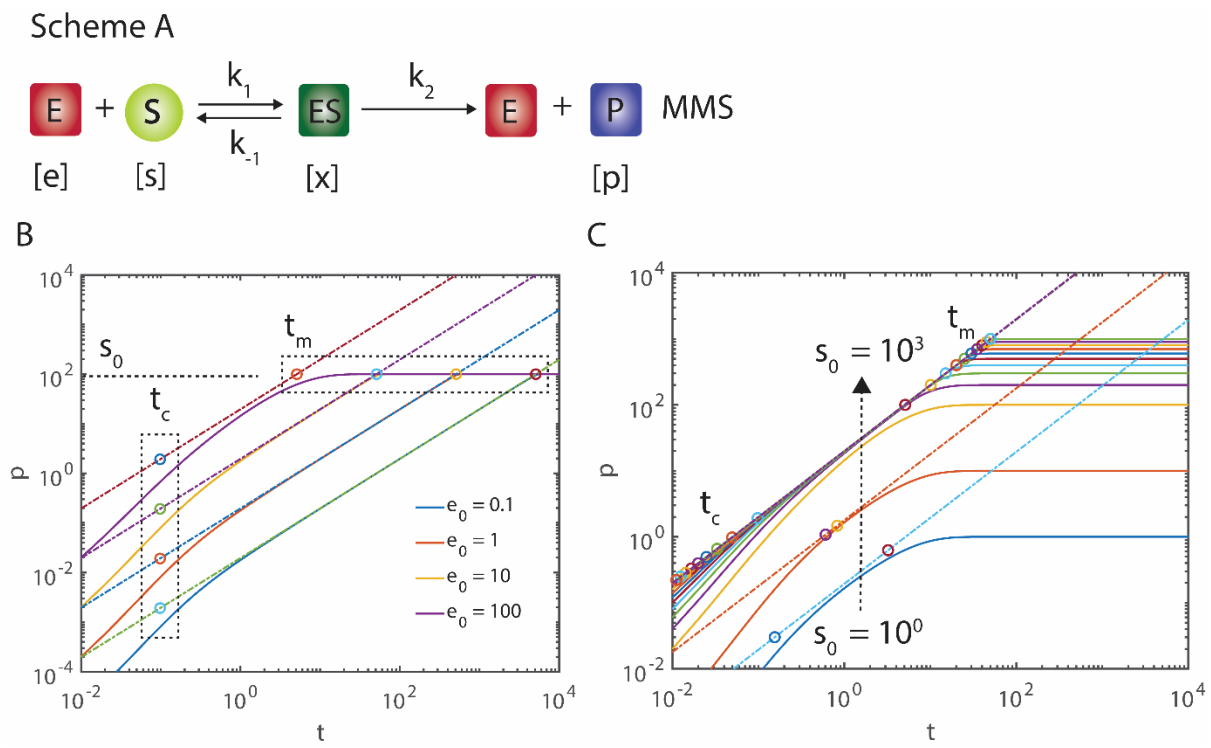


**FIGURE 1**. Dynamics of product formation (p, µM) in Michaelis-Menten scheme of single substrate enzyme catalyzed reactions. Here $t_c \cong 1/\left(k_1(K_M + s_0)\right)$ , $t_m \cong ((K_M + s_0)/k_2 e_0)$ are the approximate pre-steady state and reaction-end timescales (t, seconds) and numerical integration settings from **Eqs. 1** are $k_1$ = 0.1 µM$^{-1}$s$^{-1}$, $k_{-1}$ = 0.01 s$^{-1}$ and $k_2$ = 0.2 s$^{-1}$, with $K_M$ = 2.1 µM. **A**. $s_0$ = 100 µM and $e_0$ was iterated in (0.1, 1, 10, 100) µM. **B**. $e_0$ = 100 µM and $s_0$ was iterated in (1, 10, 100-1000, with increment of 100) µM along the arrow.

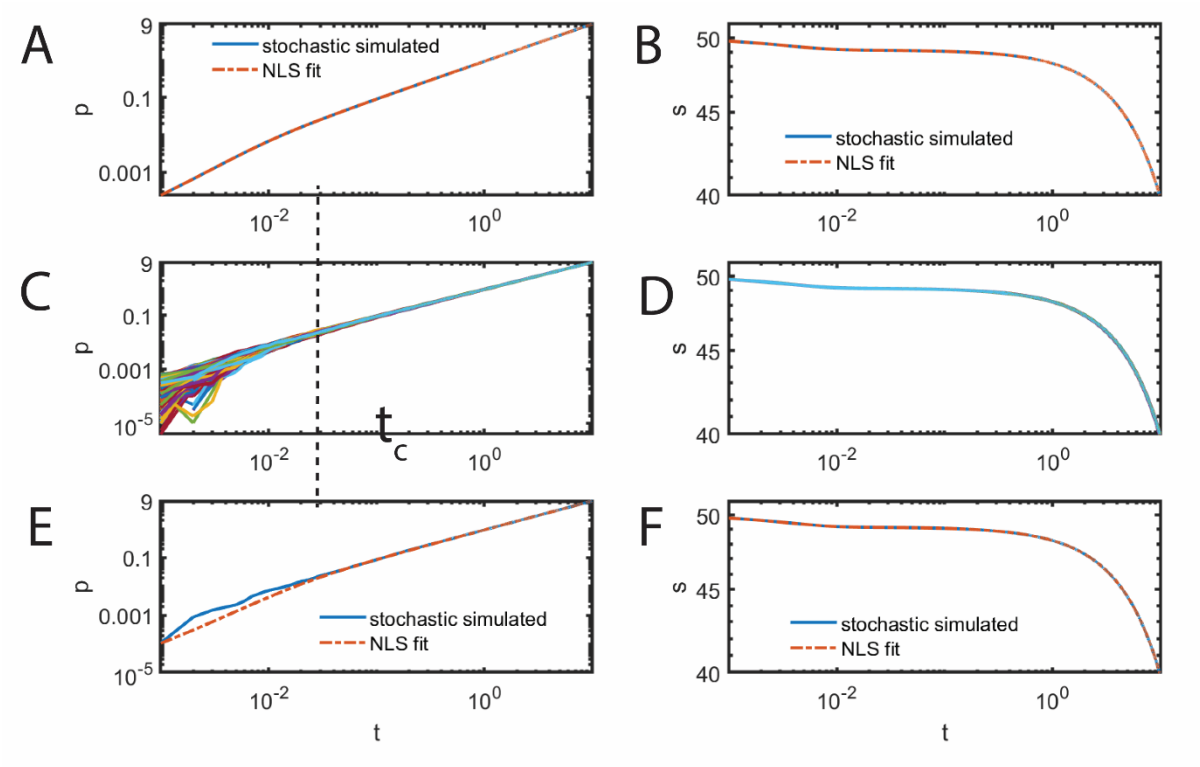


**FIGURE 2**. Progress curve analysis of data on (p, t) and (s, t). Common simulation settings for the numerical integration scheme in **Eqs. 13** are $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = $10^{-3}$ s, $e_0$ = 0.85 μM, $s_0$ = 50 μM so that $K_M$ = 0.313725 μM, $t_c \approx$ 0.0039 s, $t_m \approx$ 118.38 s, total simulation time = 10 s and $v_{max}$ = 0.935 μM $s^{-1}$. **A**. Simulation of **Eqs. 13** with $\rho$ = 0 and NLS fit of the simulation trajectory with **Eqs. 1** using ML algorithm yielded the parameters $k_1$ = 5.1 ± 1.25 x $10^{-5}$, $k_{-1}$ = 0.499 ± 7 x $10^{-6}$, $k_2$ = 1.1 ± 4.5 x $10^{-8}$ for the data on (p, t) with $K_M$ = 0.314 at 95% confidence level. **B**. NLS fit using ML algorithm over **Eqs. 1** yielded $k_1$ = 5.1 ± 6 x $10^{-6}$, k-1 = 0.499 ± 9 x $10^{-6}$, $k_2$ = 1.1 ± 4.5 x $10^{-8}$ for data on (s, t) generated using ρ = 0 in **Eqs. 13** with $K_M$ = 0.313639 at 95% confidence level. **C**. stochastic trajectories (1000 numbers) of **Eqs. 13** with ρ = $10^{-2}$ demonstrating the fluctuations in the pre-steady state regime of (p, t) space. **D**. stochastic trajectories of **Eqs. 13** with ρ = $10^{-2}$ demonstrating less fluctuations in the pre-steady state regime of (s, t) space. **E**. NLS fit of sample (p, t) trajectory with ρ = $10^{-2}$ yielded $k_1$ = 2.29 ± 0.09, $k_{-1}$ = 1.53 ± 0.1, $k_2$ = 1.12 ± 2 x $10^{-3}$ with $K_M$ = 1.18 and $v_{max}$ = 0.95 at 95% confidence level. **F**. NLS fit of sample (s, t) trajectory with ρ = $10^{-2}$ over **Eqs. 1** yielded $k_1$ = 5.1 ± 0.07, $k_{-1}$ = 1.3 ± 0.1, $k_2$ = 1.1 ± 5 x $10^{-4}$ with $K_M$ = 0.47 and $v_{max}$ 0.94 at 95% confidence level.

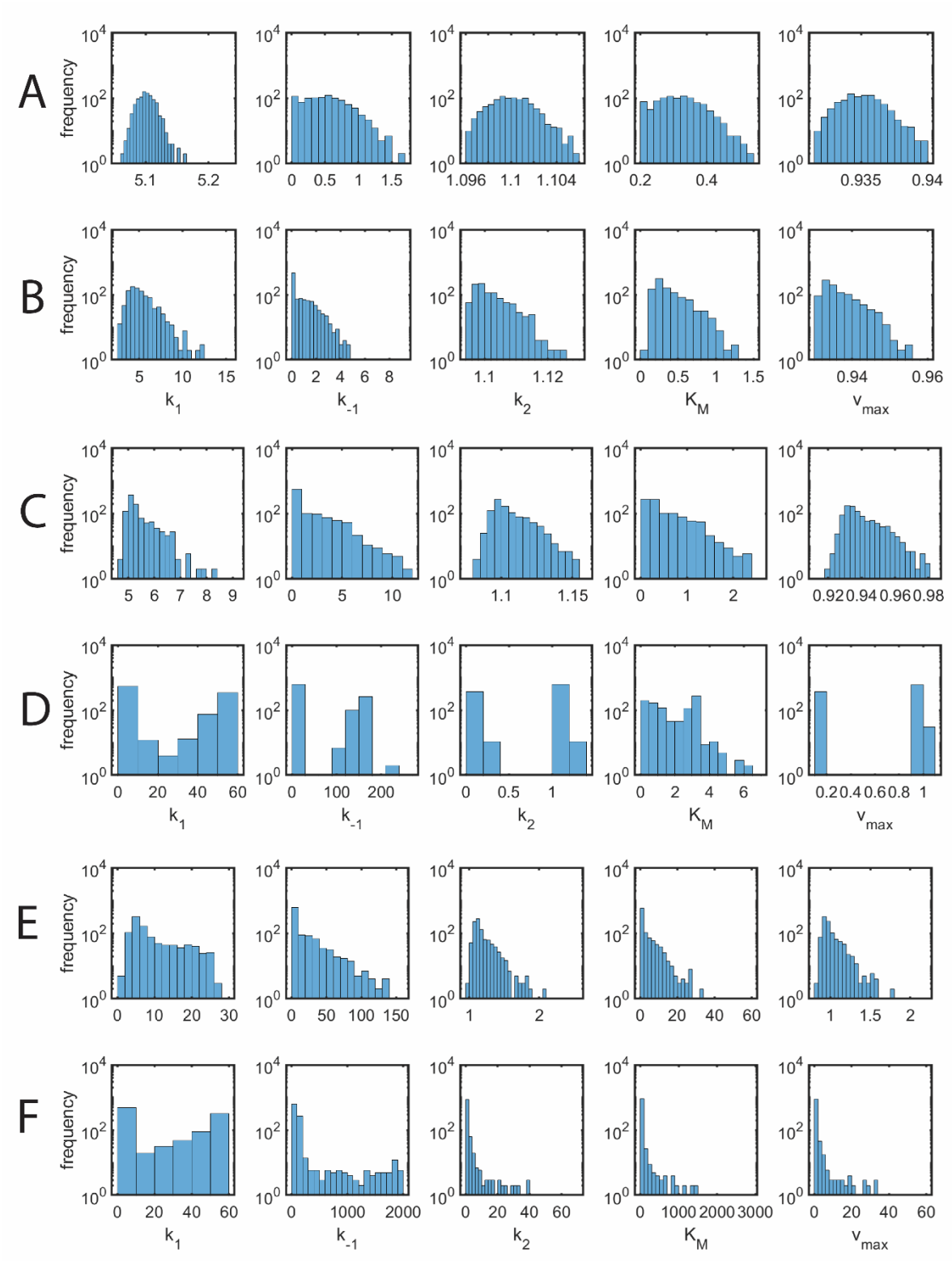


**FIGURE 3**. Distribution of nonlinear least squares (NLS) fit parameters across various noise levels. Common simulation settings for the numerical integration scheme in **Eqs. 13** are $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = $10^{-3}$ s, total simulation time = 10 s, $e_0$ = 0.85 μM so that $K_M$ = 0.313725 μM and $v_{max}$ = 0.935 μM $s^{-1}$, $s_0$ = 50 μM for which one finds that $t_c \approx 0.0039$ s and $t_m \approx 118.38$ s. NLS fit was done using Marquardt-Levenberg algorithm as outlined in the

methods section. In A, C and E, NLS fittings were done using data on (s, t). Whereas, in B, D and F the fittings were done using data on (p, t). Upon obtaining $k_1$, $k_{-1}$ and $k_2$, $K_M$ and $v_{max}$ will be calculated. Noise levels in **Eqs. 13** were set as follows. **A**, **B**. $\rho = 10^{-3}$. **C**, **D**. $\rho = 10^{-2}$. **E**, **F**. $\rho = 10^{-1}$. The median values of $K_M$ and $v_{max}$ were computed across $10^3$ replications. **A**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (5.1005, 0.5009, 1.100, 0.3140, 0.9350). **B**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (4.9188, 0.3849, 1.1001, 0.3163, 0.9351). **C**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (5.2031, 0.5499, 1.1021, 0.3197, 0.9368). **D**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (4.7469, 1.1193, 1.0981, 1.5888, 0.9334). **E**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (6.5060, 0.4776, 1.1348, 0.3727, 0.9646). **F**. the median values of ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (16.0345, 0.7332, 1.1380, 3.1860, 0.9673).

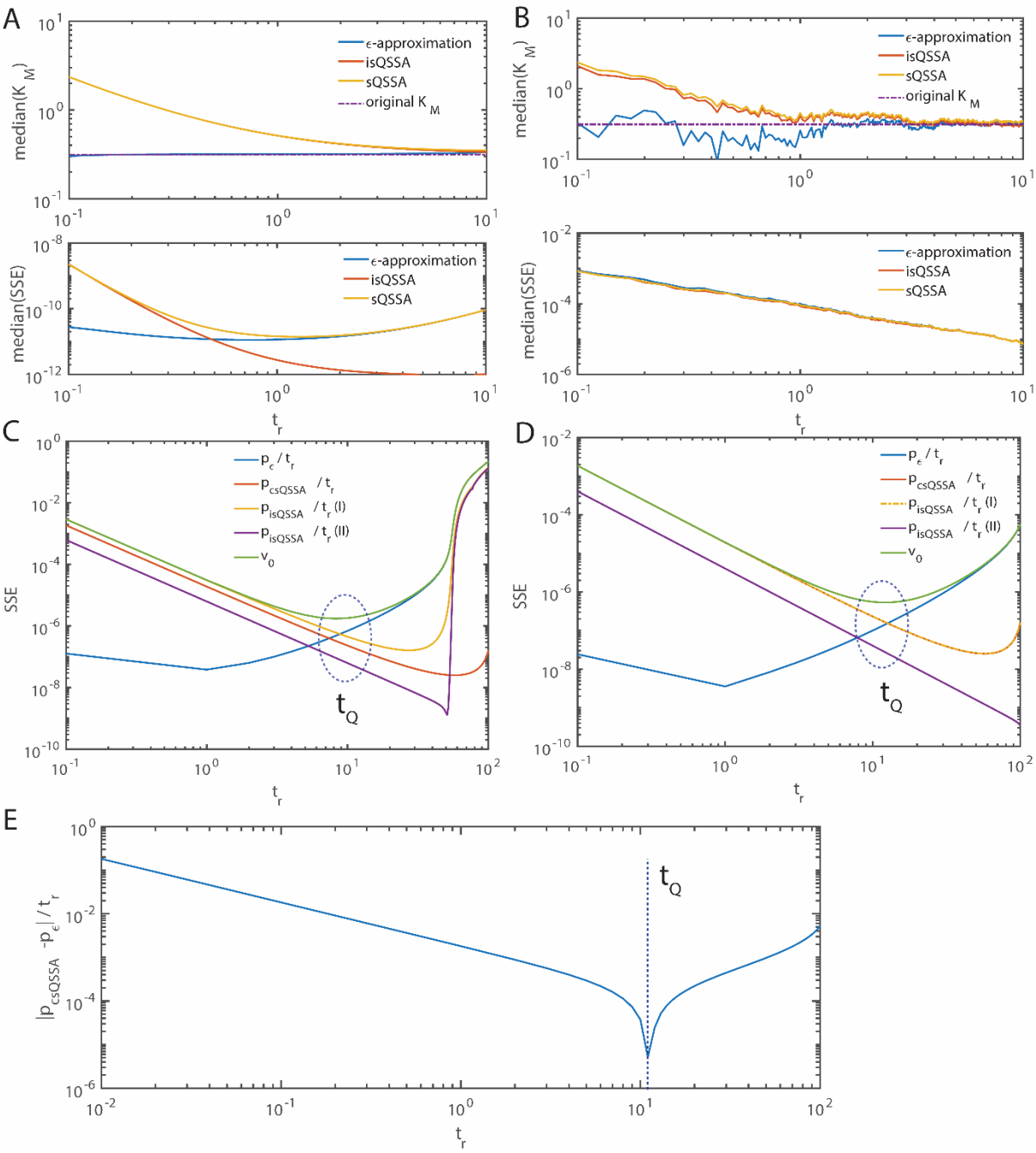


**FIGURE 4**. Variation of the fitted $K_M$ values with respect to reaction time $t_r$ across different fitting methods. Common simulation settings for the numerical integration scheme in **Eqs. 13** are $k_1$ = 5.1 μM$^{-1}$ s$^{-1}$, $k_{-1}$ = 0.5 s$^{-1}$, $k_2$ = 1.1 s$^{-1}$, $\Delta t$ = 10$^{-3}$ s, total simulation time = 10 s, $e_0$ = 0.85 μM so that $K_M$ = 0.313725 μM and $v_{max}$ = 0.935 μM s$^{-1}$, $s_0$ was iterated from 50 to 250 μM with interval of 25 μM. For $s_0$ = 50 μM one finds that $t_c \approx$ 0.0039 s and $t_m \approx$ 118.38 s. The reaction time $t_r$ was iterated from 0.1 to 10 s. The sub datasets on ($\bar{v}$, $p$, $s_0$) at different $t_r$ were generated from the main progress curve dataset on (p, t) and used to fit **Eqs. 10** (ε-approximation), **Eq. 11** (isQSSA) and $v_0$ (sQSSA) and the median values of $K_M$ were obtained across 10$^3$ replications. **A**. ρ = 0 in **Eqs. 13**. **B**. ρ = 10$^{-2}$ in **Eqs. 13**. **C**. $\Delta t$ = 10$^{-5}$ s, $s_0$ was iterated from 50 to 250 μM with interval of 25 μM. For $p_{csQSSA}$ / $t_r$, $s_0$ was iterated from 100 to 200 μM with interval of 10 μM. **D**. For all the steady state methods, $\Delta t$ = 10$^{-5}$ s, $s_0$ was iterated from 100 to 200 μM with interval of 10 μM. **E**. $s_0$ = 100 μM and $\Delta t$ = 10$^{-5}$ s for which $t_Q \approx$ 10.49 s.

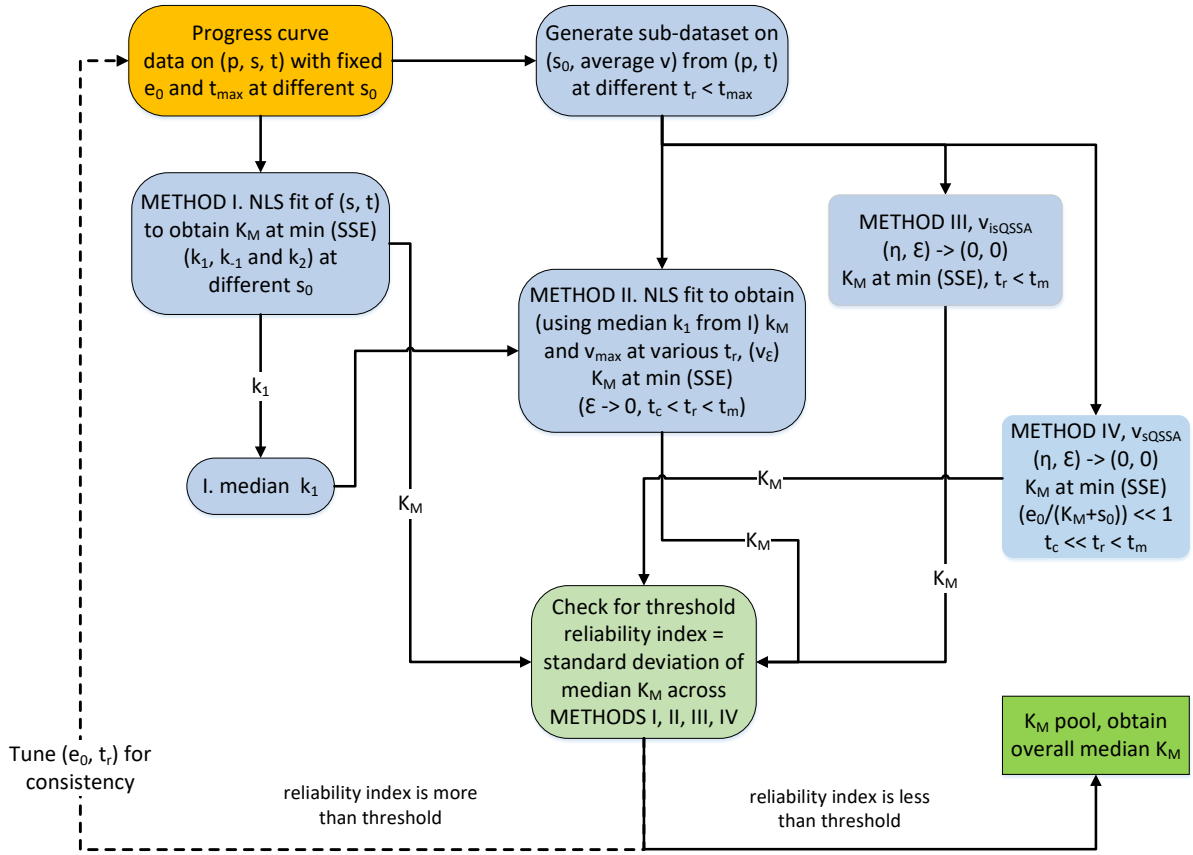


**FIGURE 5**. Computational workflow to obtain consistent estimate of $K_M$ from the progress curve dataset. Here, the progress curve data on (p, s, t) will be the main input for Methods I, II, III and IV. Here $t_{max}$ is the maximum data collection time. Method I is the direct nonlinear least square fitting of the dataset on (s, t) with fixed $e_0$ and different $s_0$ across replications with the original **Eqs. 1.1-1.4** using Marquardt-Levenberg algorithm as outlined in the methods section. The parameter $k_1$ in $\bar{v}_\varepsilon$ of **Eqs. 10** of Method II will be fixed with the median of $k_1$ values obtained from Method I. For Methods II-IV, sub-dataset on average velocity ((total product formed at the end of $t_r$) / $t_r$, where $t_r$ will be iterated from $t_c$ towards $t_m$) versus $s_0$ at different $t_r$ will be generated from the main dataset. NLS fittings over ($\bar{v}$, $p$, $s_0$) were done using models corresponding to Methods II-IV at different $t_r$. The $K_M$ value corresponding to $t_r$ with minimum sum of square error (SSE) will be considered. Upon obtaining consistency on the estimates of $K_M$ across Methods I-IV corresponding to the given threshold *reliability index*, all the $K_M$ values obtained from the same dataset using different methods will be then pooled and the overall median $K_M$ will be computed. Here *reliability index* is the coefficient of variation of median $K_M$ values obtained from Methods I-IV.

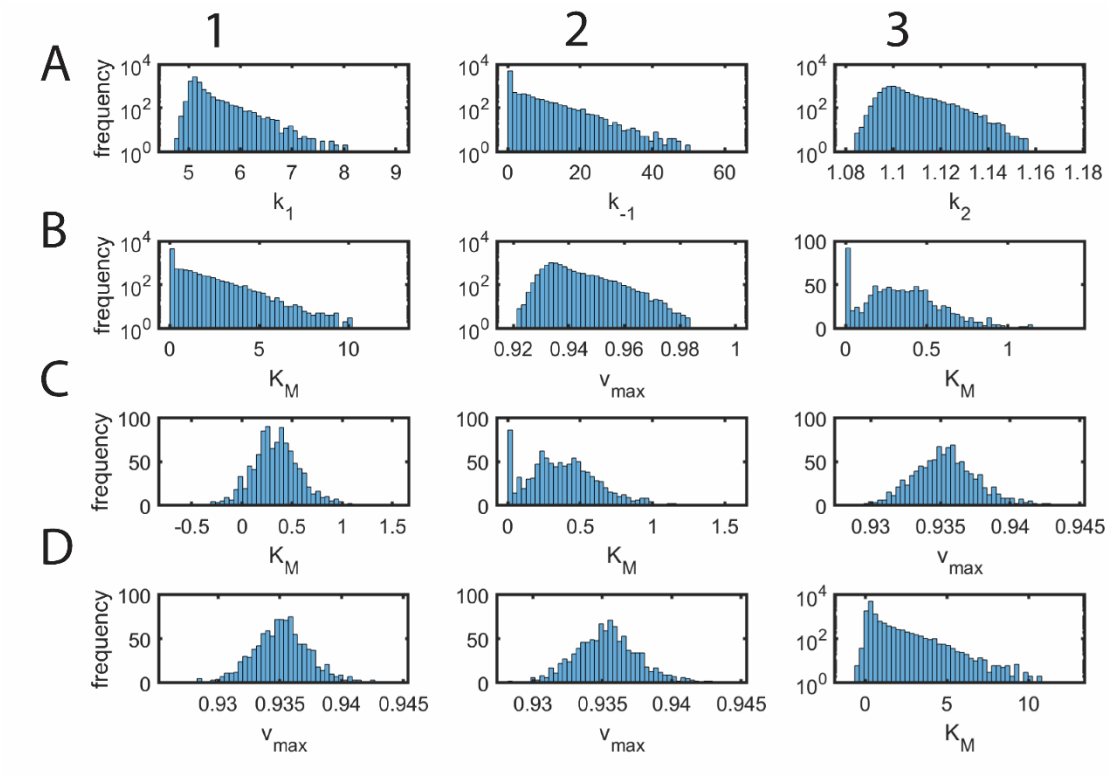


**FIGURE 6**. Distribution of fit parameters across various methods I-IV. Common simulation settings for the numerical integration scheme in **Eqs. 13** are $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = $10^{-3}$ s, total simulation time = 10 s, $e_0$ = 0.85 µM so that $K_M$ = 0.313725 µM and $v_{max}$ = 0.935 µM $s^{-1}$, $s_0$ was iterated from 50 to 250 µM with interval of 25 µM. For $s_0$ = 50 µM one finds that $t_c \approx 0.0039$ s and $t_m \approx 118.38$ s. The median values corresponding to progress curve fit over $10^3$ trajectories on (s, t) dataset were ($k_1$, $k_{-1}$, $k_2$, $K_M$, $v_{max}$) = (5.164, 0.587, 1.102, 0.331, 0.937). **A1-3, B1-2**. NLS fit method I using Marquardt-Levenberg algorithm. Upon obtaining $k_1$, $k_{-1}$ and $k_2$, $K_M$ and $v_{max}$ will be calculated as in **B1, B2**. **B3, C3**. Method II. Here the obtained median values from datasets on (p, t) were ($K_M$, $v_{max}$) = (0.338, 0.935). **C1, D1**. Method III. Here the obtained median values from datasets on (p, t) were ($K_M$, $v_{max}$) = (0.332, 0.935). **C2, D2**. Method IV. Here the obtained median values from datasets on (p, t) were ($K_M$, $v_{max}$) = (0.366, 0.935). **D3**. Overall distribution of pooled $K_M$ values across all the methods I-IV with overall median of $K_M$ = 0.3418 µM which correspond to the coefficient of variation of median $K_M$ values obtained from Methods I, II, III and IV as 0.048.

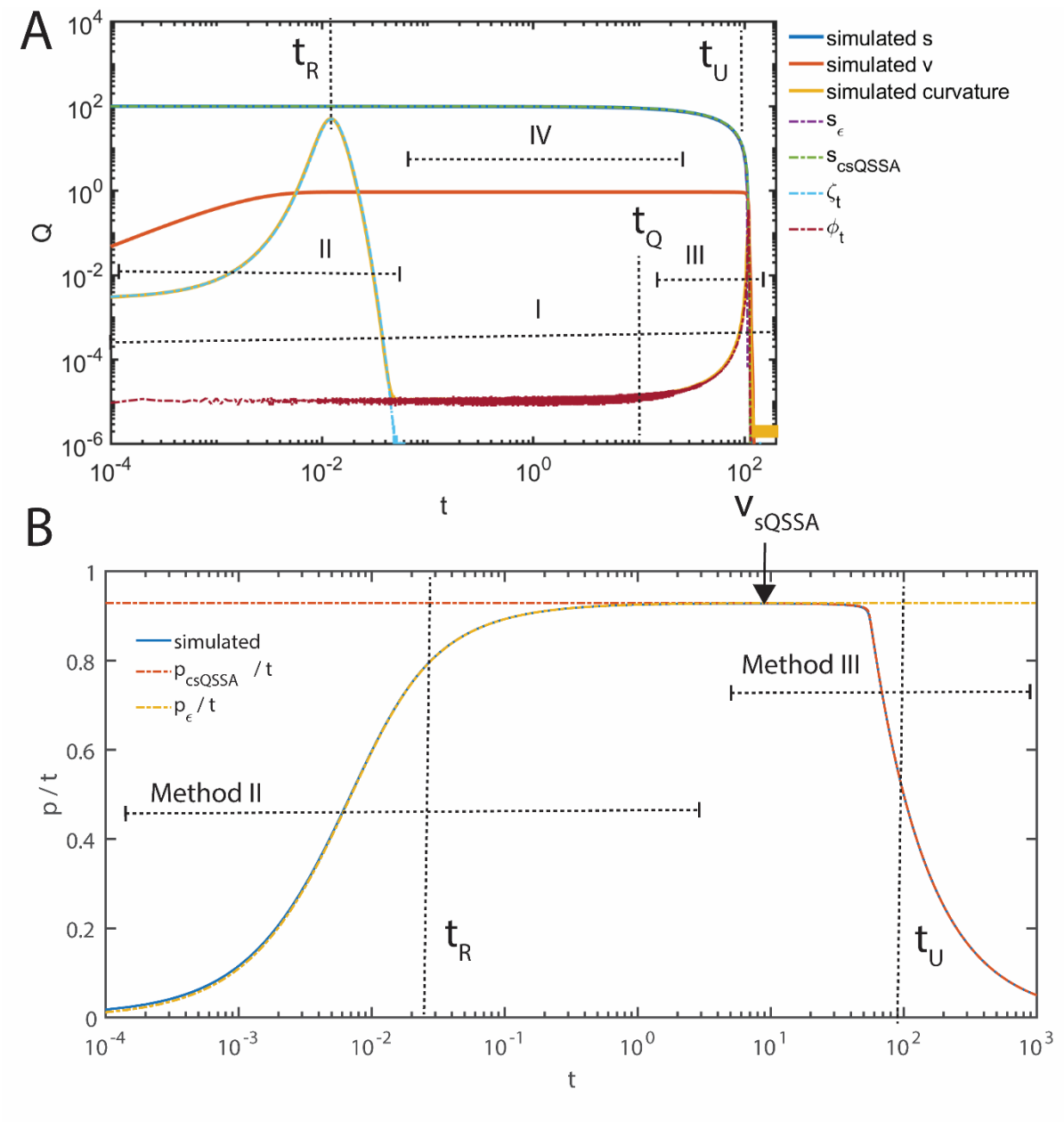


**FIGURE 7**. Change in the curvature of trajectory on (s, t) space and average reaction velocity. Common simulation settings for the numerical integration scheme in **Eqs. 13** are $k_1$ = 5.1 $\mu M^{-1} s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = 0.5 x $10^{-4}$ s, total simulation time = 1000 s, noise level set as ρ = 0, $e_0$ = 0.85 μM so that $K_M$ = 0.313725 μM and $v_{max}$ = 0.935 μM $s^{-1}$, and $s_0$ = 100 μM. For these settings, one finds that $t_c \approx 0.002$ s, $t_m \approx 91.2$ s and $t_Q \approx 10.49$ s. **A**. The transition timescales ($t_R$, $t_U$) at which maximum change in the curvature occurs can be calculated using **Eqs. 25-27** as $t_R$ = 0.0126 s and $t_U$ = 92 s. The time dependent curvatures associated with the trajectory in (s, t) space $\zeta_t$ and $\psi_t$ corresponding to the pre- and post-steady state regimes respectively are defined as in **Eqs. 24-27**. Progress curve Method I works very well in the regions with maximum change in the curvature. Whereas, steady state methods II, III and IV work very well in the regions with minimal change in the curvature. **B**. Change in the average velocity (p / t) with time. Method II works well from the pre-steady to steady state regime. Whereas, Method III works well in the steady to post-steady state regime. Method IV works well only in a narrow timescale regime with constant average velocity.

**TABLE 1. Conditions of validity of various fitting methods to obtain $K_M$ and $v_{max}$**

| Description | Definition | Validity conditions |
|---|---|---|
| Progress curve analysis | Time course dataset on (s, p, t). | timescale interval should capture both pre- and post-steady state regimes. (Method I) |
| $\bar{v}_{sQSSA}$ | $\bar{v}_{sQSSA} \cong v_{max}s_0/(K_M + s_0)$ | $t_c \ll t_r < t_m$, $(\eta, \varepsilon) \to (0,0)$, $(e_0/(K_M + s_0)) \ll 1$, Method IV |
| $\bar{v}_{isQSSA}$ | $\bar{v}_{isQSSA} \cong v_{max} + \frac{K_M}{t_r} ln\left(\frac{s_0 - p}{s_0}\right)$ (I)<br>$\bar{v}_{isQSSA} \cong v_{max} + \frac{K_M}{t_r} ln\left(\frac{s}{s_0}\right)$ (II) | $t_c < t_r < t_m$, $(\eta, \varepsilon) \to (0,0)$ Method III. |
| $\bar{v}_{csQSSA}$ | $\bar{v}_{csQSSA} \cong \frac{1}{t_r}\left(s_0 - s_{csQSSA}\right)$<br>$s_{csQSSA} \cong K_M W\left(\frac{s_0}{K_M} exp\left(\frac{s_0 - v_{max}t_r}{K_M}\right)\right)$ | $t_c < t_r < t_m$, $(\eta, \varepsilon) \to (0,0)$ Method III. Alternate form of $\bar{v}_{isQSSA}$. |
| $\bar{v}_{\varepsilon}$ | $\bar{v}_{\varepsilon} \cong v_0(1 - t_c h(t_r)/t_r)$<br>$v_0 = v_{max}s_0/(K_M + s_0)$<br>$t_c \cong 1/k_1(K_M + s_0)$<br>$h(t_r) = \left(1 - exp(-t_r/t_c)\right)$ | $t_r < t_m$, $\varepsilon \to 0$<br>Method II. median of $k_1$ obtained from Method I will be substituted in $t_c$. |

# Theory on the accurate estimation of Michaelis-Menten enzyme kinetic parameters from steady state and progress curve datasets

## SUPPORTING MATERIALS

### Datasets for the global analysis

DOI: 10.5281/zenodo.15094623

**FIGURES 2C, 2D**

sampleptr.txt and samplestr.txt

Simulation settings **(Eqs. 13**) are $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = $10^{-3}$ s, $e_0$ = 0.85 μM, $s_0$ = 50 μM and ρ = $10^{-2}$. Totally there are 1000 replications. First column is the time in seconds, and subsequent columns are concentrations of product (sampleptr.txt) and substrate (samplestr.txt) respectively.

**FIGURE 3**

Simulation settings (Eqs. 13) are $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, $\Delta t$ = $10^{-3}$ s, $e_0$ = 0.85 μM and $s_0$ = 50 μM. Totally there are 1000 replications. First column is the time in seconds, and subsequent columns are replications of concentrations of product (sampleptr.txt) and substrate (samplestr.txt) respectively.

sampledat_ptpm1.txt, sampledat_ptpm2.txt, sampledat_ptpm3.txt respectively with ρ = $10^{-1}$, $10^{-2}$ and $10^{-3}$ for product evolution.
sampledat_stpm1.txt, sampledat_stpm2.txt, sampledat_stpm3.txt respectively with ρ = $10^{-1}$, $10^{-2}$ and $10^{-3}$ for substrate evolution.

**FIGURES 2E, 2F, 4** and **6**

Datasets were generated by numerical integration of **Eqs. 13** as given in main text with the following settings $k_1$ = 5.1 $\mu M^{-1}s^{-1}$, $k_{-1}$ = 0.5 $s^{-1}$, $k_2$ = 1.1 $s^{-1}$, ρ = $10^{-2}$, $\Delta t$ = $10^{-3}$ s, $e_0$ = 0.85 μM, $s_0$ was iterated from 50 to 250 with interval 25 μM. Totally 1000 replications were generated for the global analysis as in **Fig. 5** of the main text. The data files are named as "sample_enzkin_**w**t_noise_data_R_**N**.dat" where N is the replication number taking values from 0 to 999 and w = s or p depending on the trajectory of substrate or product. The first column is time and columns 2 to 10 represents the data corresponding to different initial substrate concentrations $s_0$ viz. 50, 75, 100…250 μM. Time course sample datasets on the trajectories of substrate depletion were using in direct nonlinear least square fit with **Eqs. 1** to obtain rate constants $[k_1, k_{-1}, k_2]$ using Marquardt-Levenberg algorithm as depicted in methods section. Using these fit values $K_M$ and $v_{max}$ were calculated as $K_M = (k_{-1} + k_2)/k_1$ and $v_{max} = k_2 e_0$. Analysis over 1000 such time course datasets, one obtains 1000 such values of $[k_1, k_{-1}, k_2, K_M, v_{max}]$ which were used

to construct the histogram as depicted in **Figs. 6**. Trajectories on the product evolution were used for the steady state analysis by computing average velocities (p / t) at a given reaction time across various substate concentrations and replications.

**Fig. 2E**: "sample_enzkin_**pt**_noise_data_R_**0**.dat", first and second columns were used as time course data ($s_0$ = 50 μM) to NLS fit of (p, t).

**Fig. 2F**: "sample_enzkin_**st**_noise_data_R_**0**.dat", first and second columns were used as time course data ($s_0$ = 50 μM) to NLS fit of (s, t).

### Sample MATLAB code for NLS fit of time course (p, t) data to Eqs. 1 of main text

```
function out = enzkinfitnlsp(tdat,pdat,ez,sz,alpha)
outn = [tdat pdat];
maxt = outn(length(outn(:,1)),1);
delt = tdat(2,1)-tdat(1,1);
gx = gca;
gx.FontSize = 20;
gx.LineWidth = 2;
v = outn(:,2);
semilogx(outn(:,1),v,'LineWidth',2); hold on;
dik1 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinpt(k1+dpram,k_1,k2,ez,sz,delt,maxt)-
enzkinpt(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
dik_1 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinpt(k1,k_1+dpram,k2,ez,sz,delt,maxt)-
enzkinpt(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
dik2 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinpt(k1,k_1,k2+dpram,ez,sz,delt,maxt)-
enzkinpt(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
lambda = 1e-1; lambda_l = 1e-8; lambda_h = 1e8; tune_p = 5; dpram = 1e-4; nite = 200000; stat = 1;
dof = length(outn(:,1))-3;
param = zeros(3,1); param(1:3,1) = [0.1 1 0.1]; tol = 1e-12;
out =
struct('alpha',alpha,'params',0,'total_iterations',0,'initial_lambda',0,'RMSE',0,'final_lambda',0,'tun
e_param',0,'tpvalues',0,'errDOF',0,'totDOF',0,'regDOF',0,'MSE',0,'nparam',0,'tstat',0,'covmat',0,'SE_p
arams',0,'H0_params',0,'SST',0,'SSR',0,'SSE',0,'fcal',0,'H0_f',0,'fpvalue',0,'R_square',0,'R_square_ad
j',0);
out.initial_lambda = lambda;
out.errDOF = dof;
out.regDOF = 2;
out.totDOF = length(outn(:,1))-1;
out.nparam = 3;
for w=1:nite
    if stat == 1
        jmat = [dik1(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)';
dik_1(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)';
dik2(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)']';
        jtjm = jmat'*jmat;
        dy = v - enzkinpt(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
        if w == 1, chia = sum(dy.^2); end
    end
    for i=1:length(jtjm), jtjm(i,i) = jtjm(i,i) + jtjm(i,i) * lambda; end
    ijtjm = jtjm^-1;
    dyn = v - enzkinpt(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
    nparam = param + ijtjm * jmat'*dyn;
    dym = v - enzkinpt(nparam(1,1),nparam(2,1),nparam(3,1),ez,sz,delt,maxt);
    chib = sum(dym.^2);
    if abs(chia-chib) < tol || w == nite
        param = nparam;
        out.params = param;
        out.final_lambda = lambda;
        out.tune_param = tune_p;
        SSE = chib; SST = sum(v.^2) - sum(v)^2/length(v);
        out.SSE = SSE; out.SST = SST; out.SSR = out.SST-out.SSE;
        out.fcal = (out.SSR/out.regDOF)/(out.SSE/out.errDOF);
        out.fpvalue = fcdf(out.fcal,out.regDOF,out.errDOF,'upper');
        if out.fpvalue < alpha
            out.H0_f = 1;
        else
```

```
            out.H0_f = 0;
        end
        SSEr = SSE/dof; RMSE = sqrt(SSEr);
        rsquare = 1 - SSE/SST;
        out.R_square = rsquare;
        rsqadj = 1-((length(v)-1)/dof) * (1-rsquare);
        out.R_square_adj = rsqadj;
        cov_mat = ijtjm * SSEr;
        out.covmat = cov_mat;
        param_err = zeros(3,2);
        out.MSE = out.SSE/out.errDOF;
        param_err(:,1) = param - (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        param_err(:,2) = param + (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        out.tstat = param.*sqrt(diag(cov_mat)).^-1;
        out.tpvalues = zeros(3,1); out.H0_params = zeros(3,1);
        for ip=1:3
            out.tpvalues(ip,1) = tpval(out.tstat(ip,1),out.errDOF);
            if out.tpvalues(ip,1) < alpha
                out.H0_params(ip,1) = 1;
            else
                out.H0_params(ip,1) = 0;
            end
        end
        SET = (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        out.SE_params = param_err;
        disp(['param (k1, k_1, k2), ' 'param_low, ' 'param_high, ' 'Std. Err']);
        disp([param param_err SET]);
        disp(['total iterations, '  'SSE, ' 'RMSE, ' 'rsquare, ' 'rsqadj, ' 'lambda']);
        disp([w SSE RMSE rsquare rsqadj lambda]);
        out.total_iterations = w;
        out.RMSE = RMSE;
        break;
    end
    if chib < chia
        if lambda/tune_p >= lambda_l, lambda = lambda/tune_p; end
        param = nparam;
        chia = chib;
        stat = 1;
    else
        if lambda * tune_p <= lambda_h, lambda = lambda * tune_p; end
        stat = 0;
    end
end
% get the NLS fit dataset
qpt = enzkinpt(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
semilogx(outn(:,1),qpt,'LineWidth',2,'MarkerSize',10);
hold off;
legend('Simulation','NLS fit');
legend('boxoff');
xlabel('t'); ylabel('p');
pt = gca; pt.FontSize = 20;
xscale('log');
yscale('log');
end

function out = enzkinpt(k1,k_1,k2,ez,sz,delt,tot)
nt = ceil(tot/delt);
out = zeros(nt,1);
t = 0; pt = 0; xt = 0; i = 1;
while t <= tot
    out(i,1) = pt;
    xt = xt + delt * (k1*(ez-xt)*(sz-xt-pt)-(k_1+k2)*xt);
    pt = pt + delt * k2 * xt;
    t = t + delt;
    i = i + 1;
end
end
```

## Sample MATLAB code for NLS fit of time course (s, t) data to Eqs. 1 of main text

```
function out = enzkinfitnlss(tdat,sdat,ez,sz,alpha)
outn = [tdat sdat];
maxt = outn(length(outn(:,1)),1);
delt = tdat(2,1)-tdat(1,1);
gx = gca;
gx.FontSize = 20;
gx.LineWidth = 2;
v = outn(:,2);
semilogx(outn(:,1),v,'LineWidth',2); hold on;
dik1 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinst(k1+dpram,k_1,k2,ez,sz,delt,maxt)-
enzkinst(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
dik_1 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinst(k1,k_1+dpram,k2,ez,sz,delt,maxt)-
enzkinst(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
dik2 = @(k1,k_1,k2,dpram,ez,sz,maxt,delt) ((enzkinst(k1,k_1,k2+dpram,ez,sz,delt,maxt)-
enzkinst(k1,k_1,k2,ez,sz,delt,maxt))/dpram);
lambda = 1e-1; lambda_l = 1e-8; lambda_h = 1e8; tune_p = 5; dpram = 1e-4; nite = 200000; stat = 1;
dof = length(outn(:,1))-3;
param = zeros(3,1); param(1:3,1) = [0.1 1 0.1]; tol = 1e-12;
out =
struct('alpha',alpha,'params',0,'total_iterations',0,'initial_lambda',0,'RMSE',0,'final_lambda',0,'tun
e_param',0,'tpvalues',0,'errDOF',0,'totDOF',0,'regDOF',0,'MSE',0,'nparam',0,'tstat',0,'covmat',0,'SE_p
arams',0,'H0_params',0,'SST',0,'SSR',0,'SSE',0,'fcal',0,'H0_f',0,'fpvalue',0,'R_square',0,'R_square_ad
j',0);
out.initial_lambda = lambda;
out.errDOF = dof;
out.regDOF = 2;
out.totDOF = length(outn(:,1))-1;
out.nparam = 3;
for w=1:nite
    if stat == 1
        jmat = [dik1(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)';
dik_1(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)';
dik2(param(1,1),param(2,1),param(3,1),dpram,ez,sz,maxt,delt)']';
        jtjm = jmat'*jmat;
        dy = v - enzkinpt(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
        if w == 1, chia = sum(dy.^2); end
    end
    for i=1:length(jtjm), jtjm(i,i) = jtjm(i,i) + jtjm(i,i) * lambda; end
    ijtjm = jtjm^-1;
    dyn = v - enzkinst(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
    nparam = param + ijtjm * jmat'*dyn;
    dym = v - enzkinst(nparam(1,1),nparam(2,1),nparam(3,1),ez,sz,delt,maxt);
    chib = sum(dym.^2);
    if abs(chia-chib) < tol || w == nite
        param = nparam;
        out.params = param;
        out.final_lambda = lambda;
        out.tune_param = tune_p;
        SSE = chib; SST = sum(v.^2) - sum(v)^2/length(v);
        out.SSE = SSE; out.SST = SST; out.SSR = out.SST-out.SSE;
        out.fcal = (out.SSR/out.regDOF)/(out.SSE/out.errDOF);
        out.fpvalue = fcdf(out.fcal,out.regDOF,out.errDOF,'upper');
        if out.fpvalue < alpha
            out.H0_f = 1;
        else
            out.H0_f = 0;
        end
        SSEr = SSE/dof; RMSE = sqrt(SSEr);
        rsquare = 1 - SSE/SST;
        out.R_square = rsquare;
        rsqadj = 1-((length(v)-1)/dof) * (1-rsquare);
        out.R_square_adj = rsqadj;
        cov_mat = ijtjm * SSEr;
        out.covmat = cov_mat;
        param_err = zeros(3,2);
        out.MSE = out.SSE/out.errDOF;
        param_err(:,1) = param - (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        param_err(:,2) = param + (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        out.tstat = param.*sqrt(diag(cov_mat)).^-1;
        out.tpvalues = zeros(3,1); out.H0_params = zeros(3,1);
        for ip=1:3
```

```
            out.tpvalues(ip,1) = tpval(out.tstat(ip,1),out.errDOF);
            if out.tpvalues(ip,1) < alpha
                out.H0_params(ip,1) = 1;
            else
                out.H0_params(ip,1) = 0;
            end
        end
        SET = (abs(tinv(alpha/2,dof)) * sqrt(diag(cov_mat)));
        out.SE_params = param_err;
        disp(['param (k1, k_1, k2), ' 'param_low, ' 'param_high, ' 'Std. Err']);
        disp([param param_err SET]);
        disp(['total iterations, '  'SSE, ' 'RMSE, ' 'rsquare, ' 'rsqadj, ' 'lambda']);
        disp([w SSE RMSE rsquare rsqadj lambda]);
        out.total_iterations = w;
        out.RMSE = RMSE;
        break;
    end
    if chib < chia
        if lambda/tune_p >= lambda_l, lambda = lambda/tune_p; end
        param = nparam;
        chia = chib;
        stat = 1;
    else
        if lambda * tune_p <= lambda_h, lambda = lambda * tune_p; end
        stat = 0;
    end
end
% get the NLS fit dataset
qpt = enzkinst(param(1,1),param(2,1),param(3,1),ez,sz,delt,maxt);
semilogx(outn(:,1),qpt,'LineWidth',2,'MarkerSize',10);
hold off;
legend('Simulation','NLS fit');
legend('boxoff');
xlabel('t'); ylabel('p');
pt = gca; pt.FontSize = 20;
xscale('log');
yscale('log');
end

function out = enzkinst(k1,k_1,k2,ez,sz,delt,tot)
nt = ceil(tot/delt);
out = zeros(nt,1);
t = 0; pt = 0; xt = 0; i = 1;
while t <= tot
    out(i,1) = sz-xt-pt;
    xt = xt + delt * (k1*(ez-xt)*(sz-xt-pt)-(k_1+k2)*xt);
    pt = pt + delt * k2 * xt;
    t = t + delt;
    i = i + 1;
end
end
```