

Numerical Methods in Kinetics: A Mathematica Activity

We have relied on the ability to solve analytically for the concentrations of different components in a reaction as a function of time, but this is really only meaningful for very simple reactions. The steady-state (SSA) and pre-equilibrium (PE) approximations allow us to simplify the overall rate law under certain limiting cases, typically dictated by the relative rates of different reactions in a mechanism. However, not every reaction fits neatly into one of these categories. For more complex reactions, or those for which PE and SSA are not appropriate, numerical methods can be used to solve systems of coupled differential equations.

While there are many approaches to numerical analysis, the simplest (Runge-Kutta) method involves calculating the derivative of each function with respect to time (dx/dt) at each time, t . To determine the value of x at the next time point, $t + dt$, the derivative multiplied by the time step size is added to the value $x(t)$:

$$x(t + dt) = x(t) + dt \left(\frac{dx}{dt} \right) \quad (1)$$

In this lab exploration, we will use a kinetic mechanism to determine the appropriate system of differential equations, then use Mathematica's built-in NDSolve function to calculate the values of each reactant as a function of time. This will allow us to quickly observe what can happen to the mechanism as we change initial conditions (concentrations of reactions, for example), rate constants, or forbid back reactions. From there, we will determine what rate constants would place us in the PE and SSA regimes, and confirm that those approximations are consistent with the more general approach of numerical integration.

Rather than beginning with a particular reaction, we will choose a more general case:



We will consider the reverse reaction of the second step to be negligible ($k_{2r} = 0$). Before you come to lab, then, you should solve for the rates of production of C as a function of A and B twice, once using the PE approximation, and then again using the SSA. For each, decide on what the relative sizes of the rate constants will need to be in order to make that particular approximation appropriate.

The Mathematica file that I have provided you should already be fully functional, but to be able to make changes to rate constants, etc., you will need to understand its basic structure. Feel free to add comments

(* like this *)

in the code if you would like. Follow the procedures on the page that follows and answer the questions briefly enough to convince yourself that you understand what is going on. Then, for each of the results, copy and paste the plots into the worksheet as requested, so that you can compare them to one another without needing to redo previous calculations.

1. Divide the code into "blocks", and briefly explain what the code does.

2. The first block is the assignment of variables. Variables are set at the beginning to avoid having to hunt through the code to change all instances of them when you want to change a value.

a. Why is there a maximum timestep? What does that imply about the way that the code manages the time step? What are the units here?

The maximum timestep exists to ensure that the step size remains below a threshold. It is possible that over time the numeric solution may deviate when the step size is too large. Mathematica automatically sets the step size to minimize deviations and obtain numeric results with extremely high precision. The only practical effect of keeping this value small is to slow down calculations which isn't very practical at all. If precision is what we are after I would suggest using the PrecisionGoal option rather than MaxStepSize to optimize performance.

b. What are the units of Ainitial, etc?

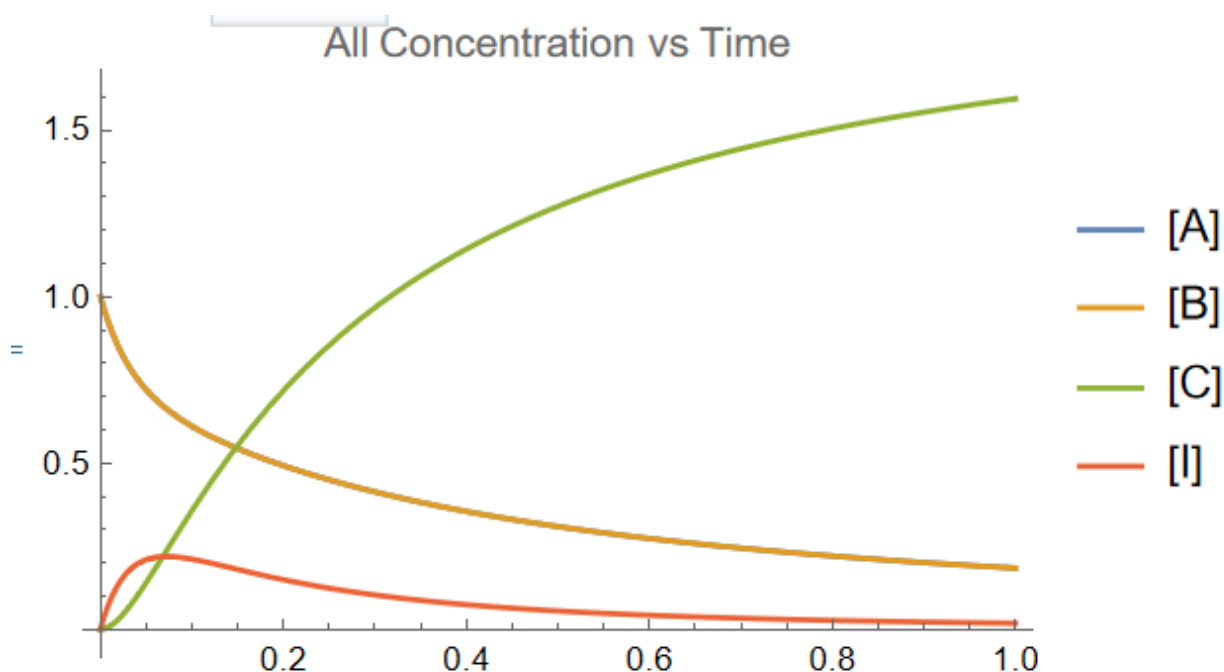
The units here are in the same time units that our rate constants use. Since SI units are typically used for these we can assume the units are seconds.

The units of the concentration initialization values are in molarity.

c. Does the set of differential equations in the data block agree with what you worked out in advance? Use it to check your work.

The differential equations used here are derived from the overall reaction.

3. Run the code. Remember, [Shift]+[Enter] must be used to execute code. This code has two parts. The first is terminated with a semicolon, so that no result will appear if it runs successfully. The second part is only "ABCModel", and will produce plots if all goes well. Once it is running, paste the plots into the space below:

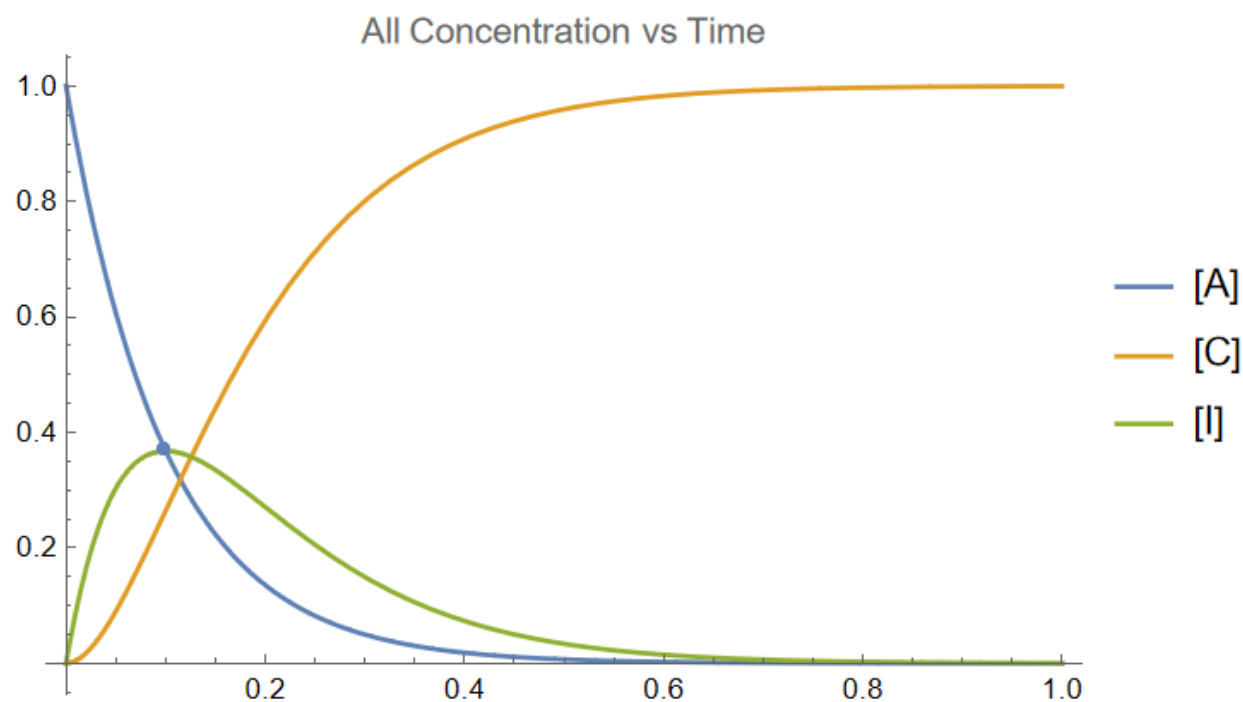


4: What changes do you see (be specific – which component changes and how) if you double the initial amount of one reactant? What happens when you then double the second reactant, keeping the first at its doubled initial concentration?

The next line allows the manipulation of the initial A concentration. If the concentration of A is brought down to 0 we notice that the reaction does not occur and B remains at 1M. If we double the concentration of A we see that the elimination rate of B is higher. As all of B is consumed, the concentration of A drops to 1M and C jumps to 2M instead of rising to an asymptote at around 1.5M. If we begin our reaction with 2M A and 2M B, I reaches its peak concentration much faster and the reaction proceeds quickly until the I peak is reached.

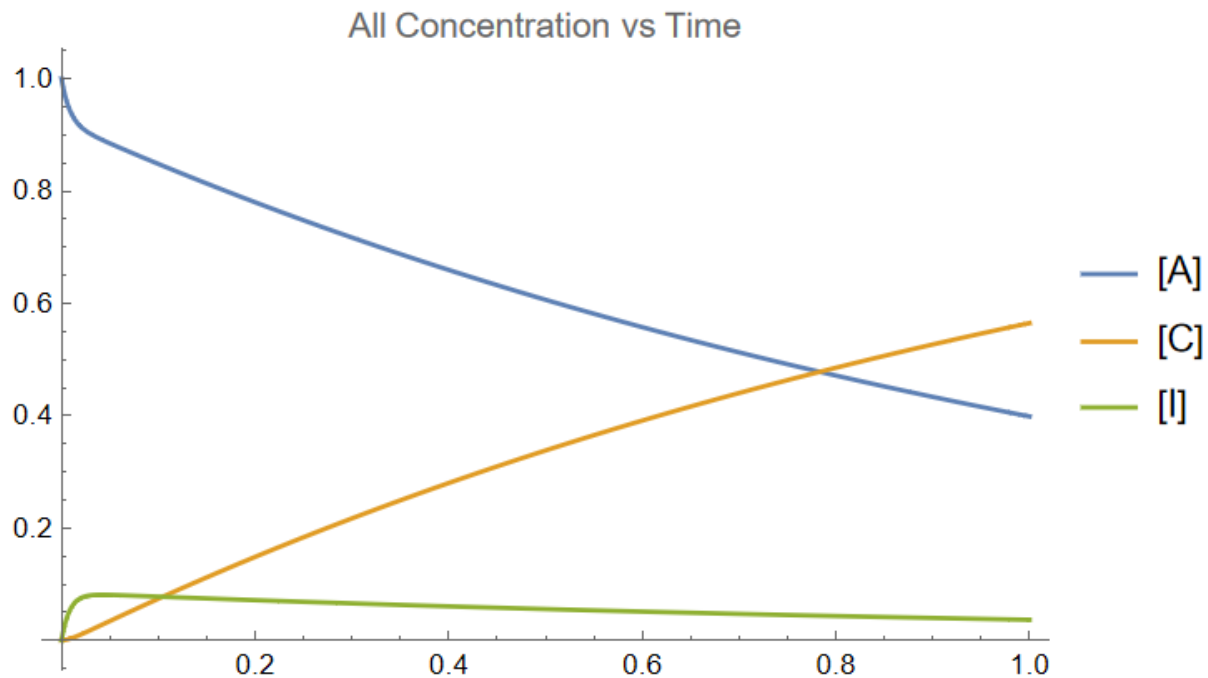
5: Reset the concentration defaults, and change the value of the rate constant for the reverse reaction in the equilibrium to be zero, effectively creating a sequential reaction. The defaults are indicated in the comments on that line. At what time does the intermediate reach its peak?

Using the reaction $A \rightarrow I \rightarrow C$:



The intermediate reaches its maximum at $t \rightarrow 0.0999991$

Now, change the value of k_1 by an order of magnitude. What happens to the time at which the intermediate peaks.

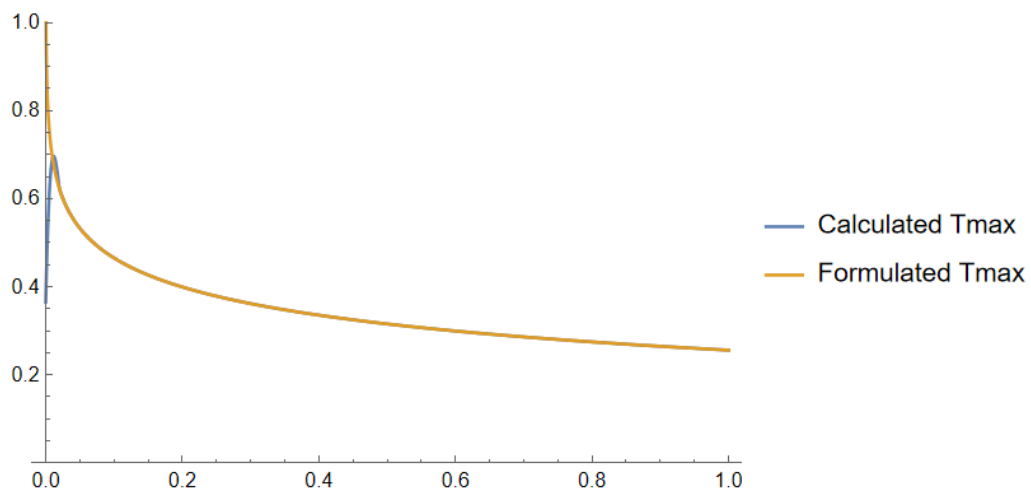


The intermediate reaches its maximum at $t \rightarrow 0.041884$

Show that these results qualitatively and quantitatively consistent with what you can calculate using

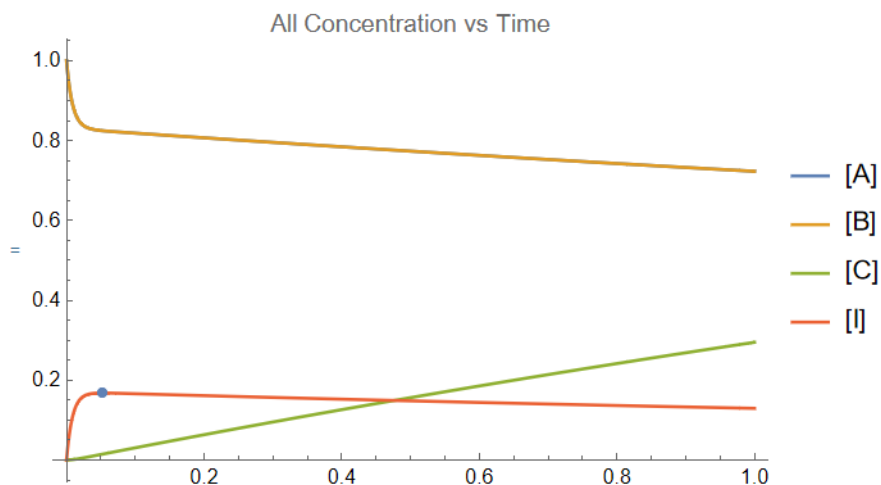
$$t_{\max} = \frac{1}{k_1 - k_2} \ln \left(\frac{k_1}{k_2} \right)$$

```
23]:= Plot[{CalculatedTmax[k1f], 1 / (k1f - k2f) Log[k1f / k2f]},
  {k1f, 0, 1}, PlotRange -> {0, 1},
  PlotLegends -> {"Calculated Tmax", "Formulated Tmax"}]
```



6: Change the order of magnitude of the rate constants (including putting k_{1r} back to a non-zero value) in a way that would reflect a PE situation. Note that you may need to make changes of orders of magnitude to accomplish your goal.

a. Paste your results here, including the rate constants you used. Is this consistent with the idea of a rapid equilibrium, followed by a slow step producing product?



The intermediate reaches its maximum at $t \rightarrow 0.0523407$

The intermediate reaches its maximum concentration at 0.167923M

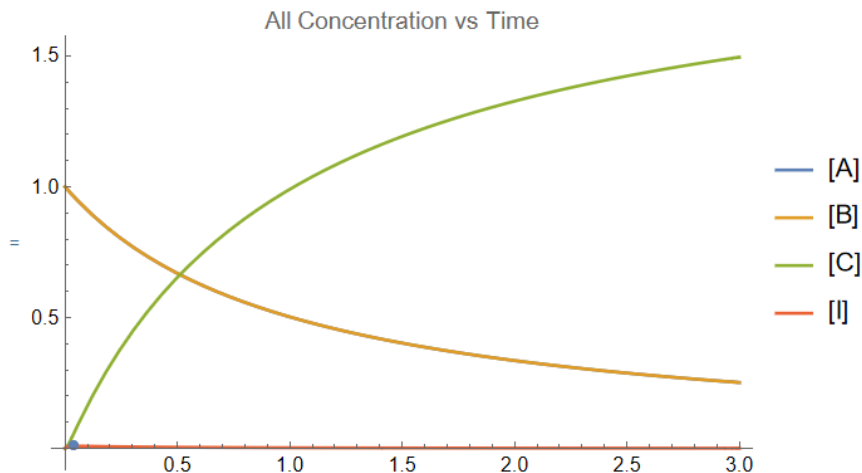
With $\{k_{1f} = 20, k_{1r} = 80, k_{2f} = 1, k_{2r} = 0\}$,
`data = NDSolve[{`

b. Does the time at which [I] peaks change as a result? How about its maximum concentration?

See graph.

7. Reconfigure the rate constants to reflect a steady-state approximation for the intermediate.

a. Paste your results, including the rate constants you used.



The intermediate reaches its maximum at $t \rightarrow 0.039991$

The intermediate reaches its maximum concentration at $0.00915953M$

b. Does the plot of $[I]$ change in a way that appears consistent with the statement that, under SSA conditions, $[I]$ is a low, nearly constant value for most of the reaction? Explain.

The steady state approximation assumes that no intermediate is formed because the formation of products is much faster than the formation of intermediate. The following situation may reflect one where the steady state approximation can be used.

One can easily see that $[I]$ remains nearly constant the entire time. The peak concentration reached is less than 0.5% of $[A]_0 + [B]_0$