Bistability in biochemical signaling models

Part 2

Icahn School of Medicine at Mount Sinai

SBCNY

Outline: Part 2

How to predict if bistability will be present?

A simple, one-dimensional example

Rate-balance plots

Ultrasensitive positive feedback can create bistability

Quantitative analyses of bistability

1) A simple "Michaelian" system

$$[A] \underset{k_{minus}}{\overset{k_{plus}}{\rightleftarrows}} [A^*]$$

A*=phosphorylated A

Total amount of [A] is constant: $[A]_{TOTAL} = [A] + [A^*]$

We want to solve for [A*] in the steady-state

$$\frac{d[A^*]}{dt} = k_{plus}\left([A]_{TOTAL} - [A^*]\right) - k_{minus}[A^*] = 0$$

$$[A^*] = \frac{k_{plus}[A]_{TOTAL}}{k_{plus} + k_{minus}}$$

or

$$\frac{[A^*]}{[A]_{TOTAL}} = \frac{1}{1 + \frac{k_{minus}}{k_{plus}}}$$

Rate balance plots

Instead of solving equations, find solution graphically

$$[A] \underset{k_{minus}}{\overset{k_{plus}}{\rightleftharpoons}} [A^*]$$

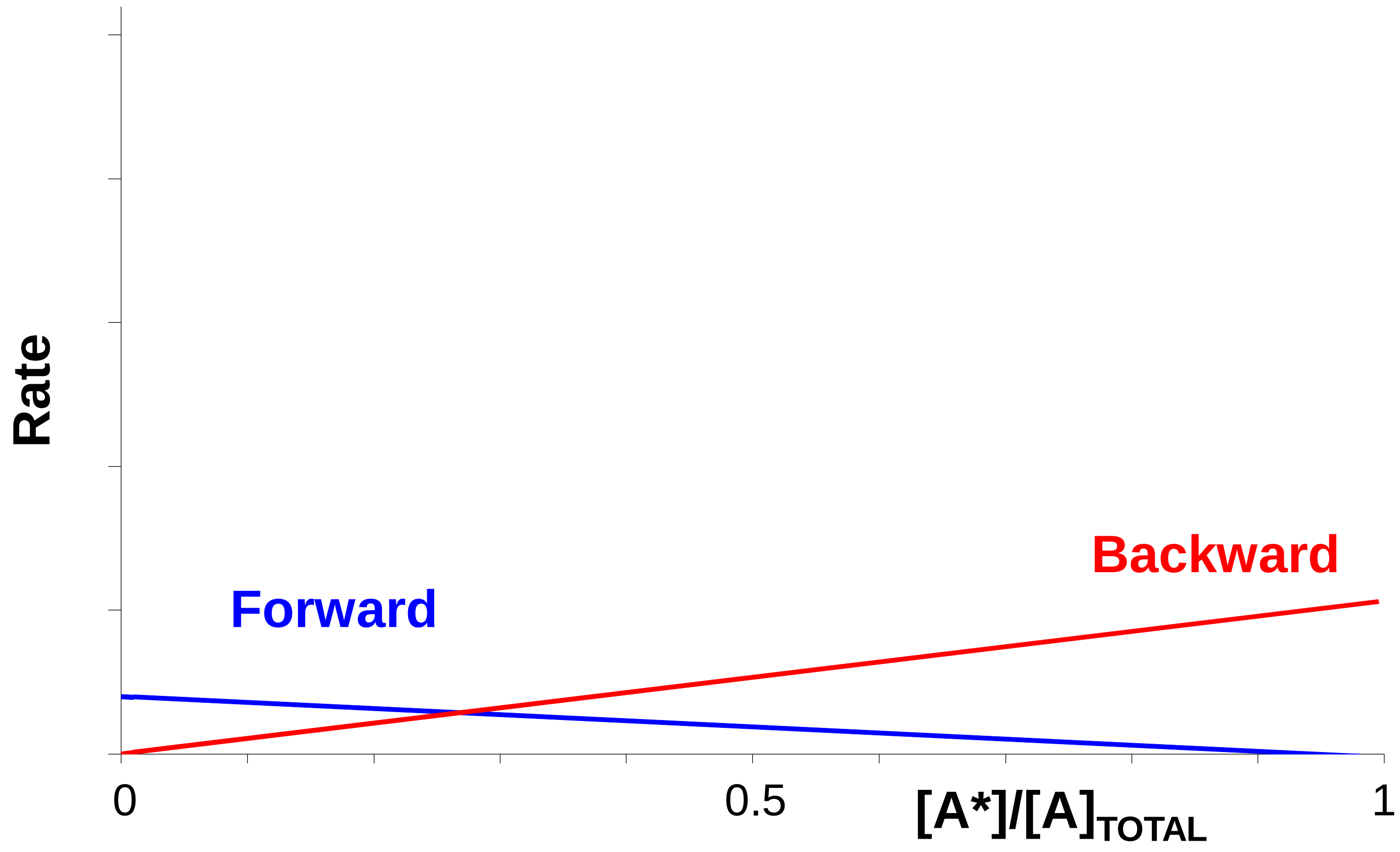
Forward Rate

$$FR = k_{plus}\left([A]_{TOTAL} - [A^*]\right)$$

Backward Rate

$$BR = k_{minus}[A^*]$$

At steady-state, FR = BR



analysis based on Ferrell & Xiong (2001) *Chaos* 11:227-236.

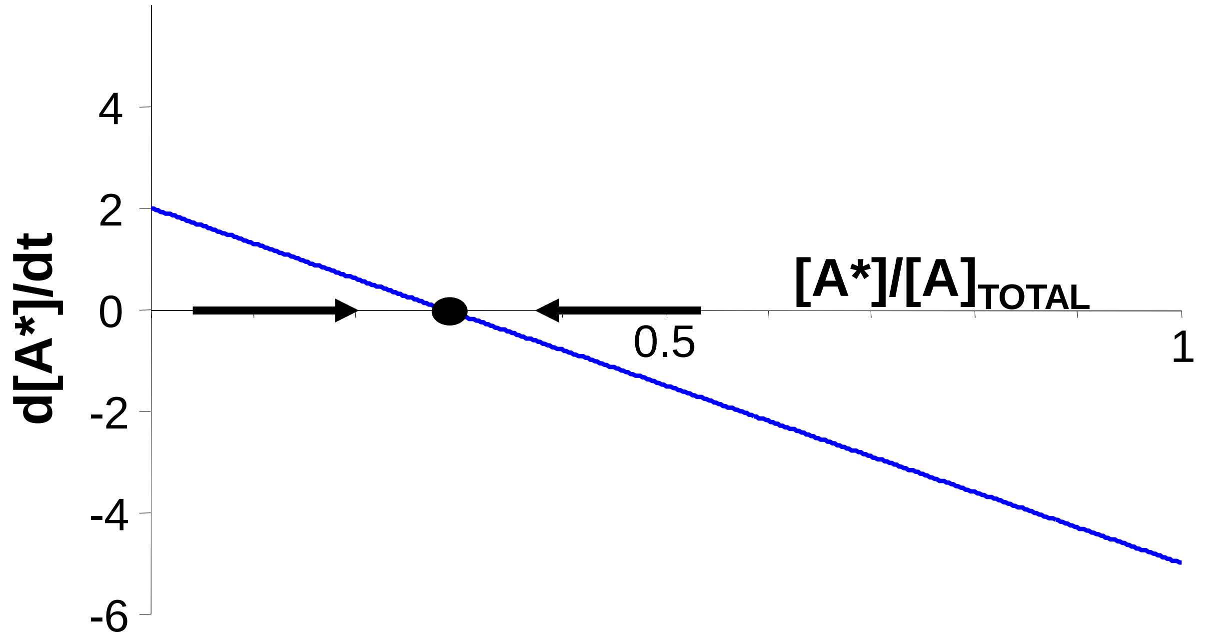
Rate balance plots

Very closely related to phase line plots

$$[A] \underset{k_{minus}}{\overset{k_{plus}}{\rightleftarrows}} [A^*]$$

$$\frac{d[A^*]}{dt} = FR - BR$$

At steady-state, FR – BR = 0



Intuitively, then, this fixed point is stable

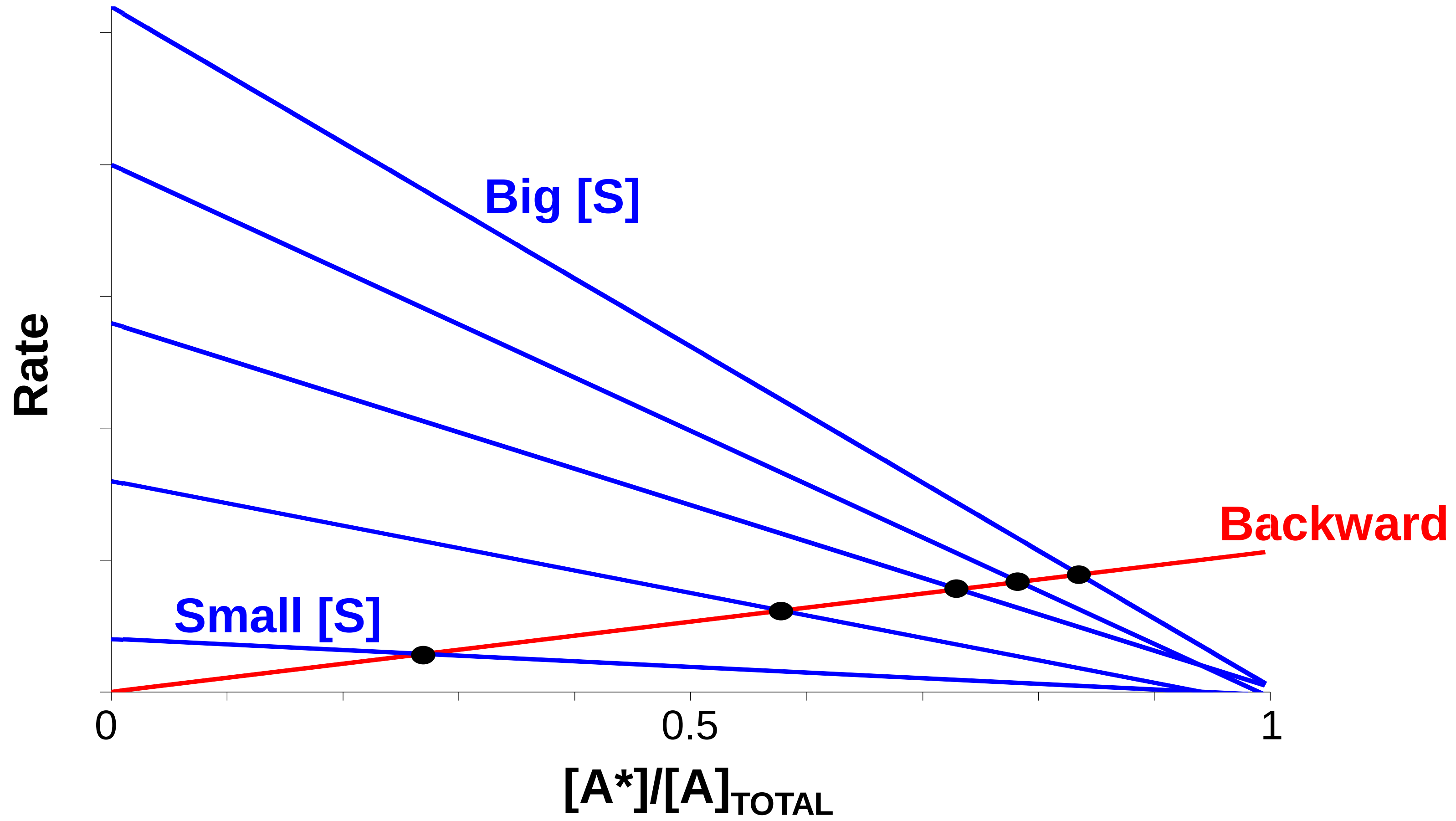
Rate balance plots

$$[A] \underset{k_{minus}}{\overset{k_{plus}}{\rightleftarrows}} [A^*]$$

Now, assume that forward rate is function of stimulus:

$$k_{plus} = k_+[S]$$

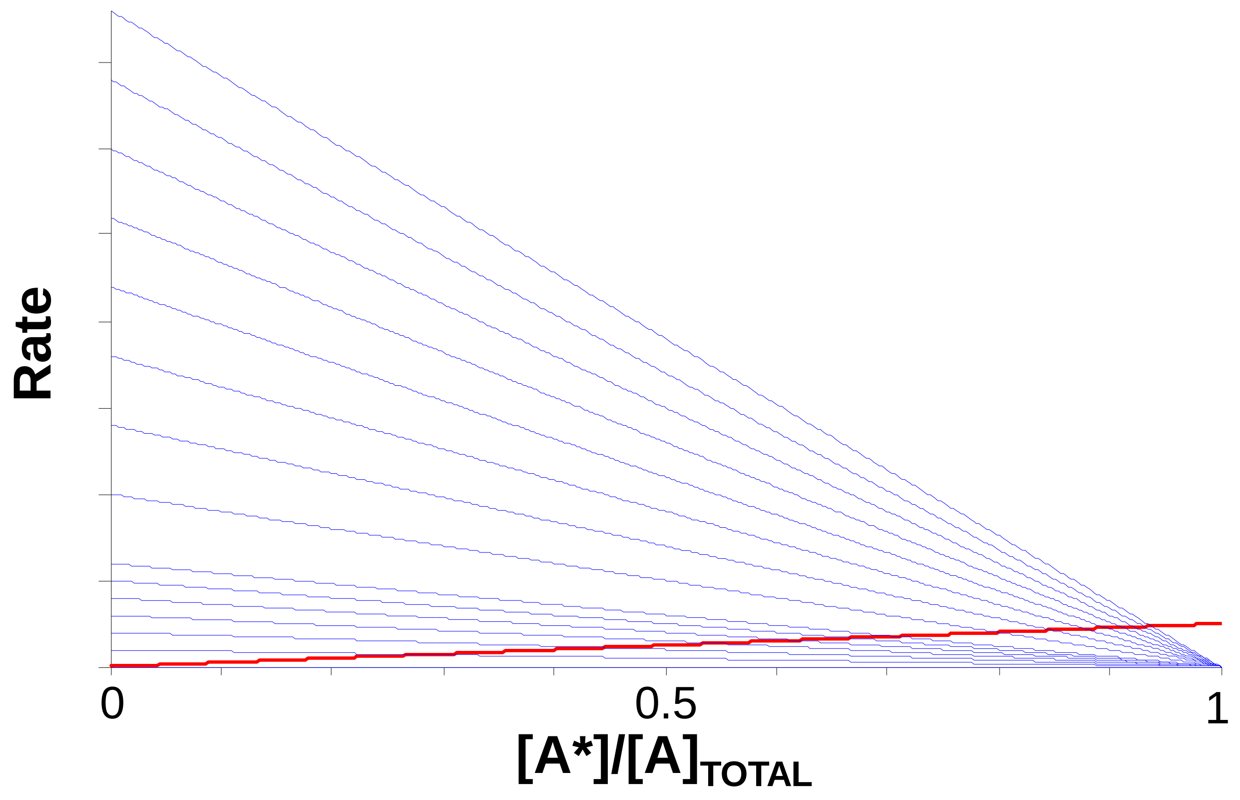
Plot rate balance for different values of stimulus [S]

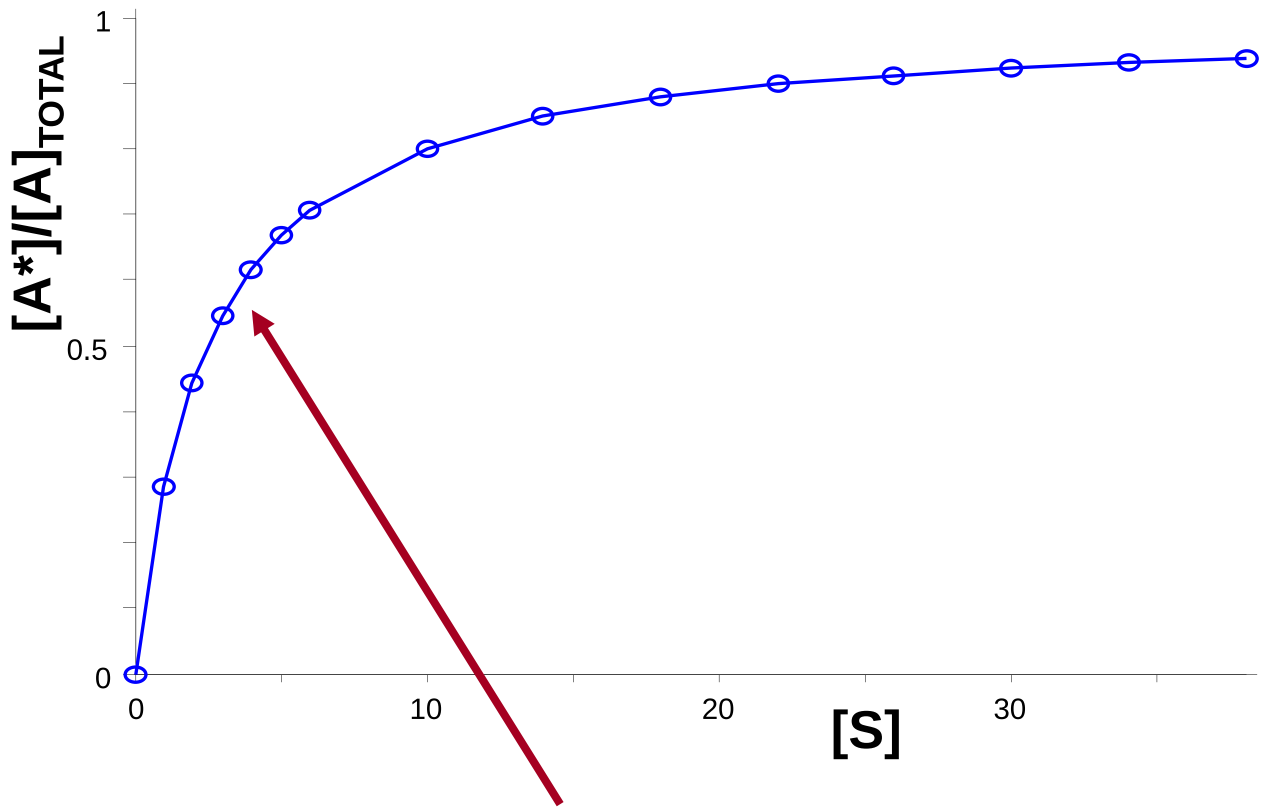


Rate balance plots

$$[A] \underset{k_{minus}}{\overset{k_+[S]}{\rightleftarrows}} [A^*]$$

This analysis can be used to plot [S] versus [A*]/$[A]_{TOTAL}$

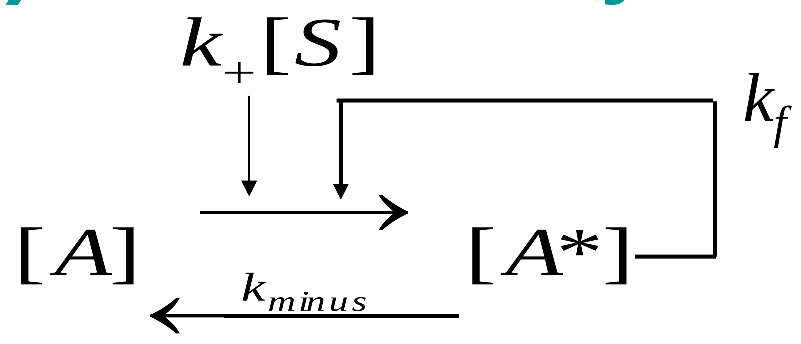




This shape is hyperbola, analogous to Michaelis-Menten equation

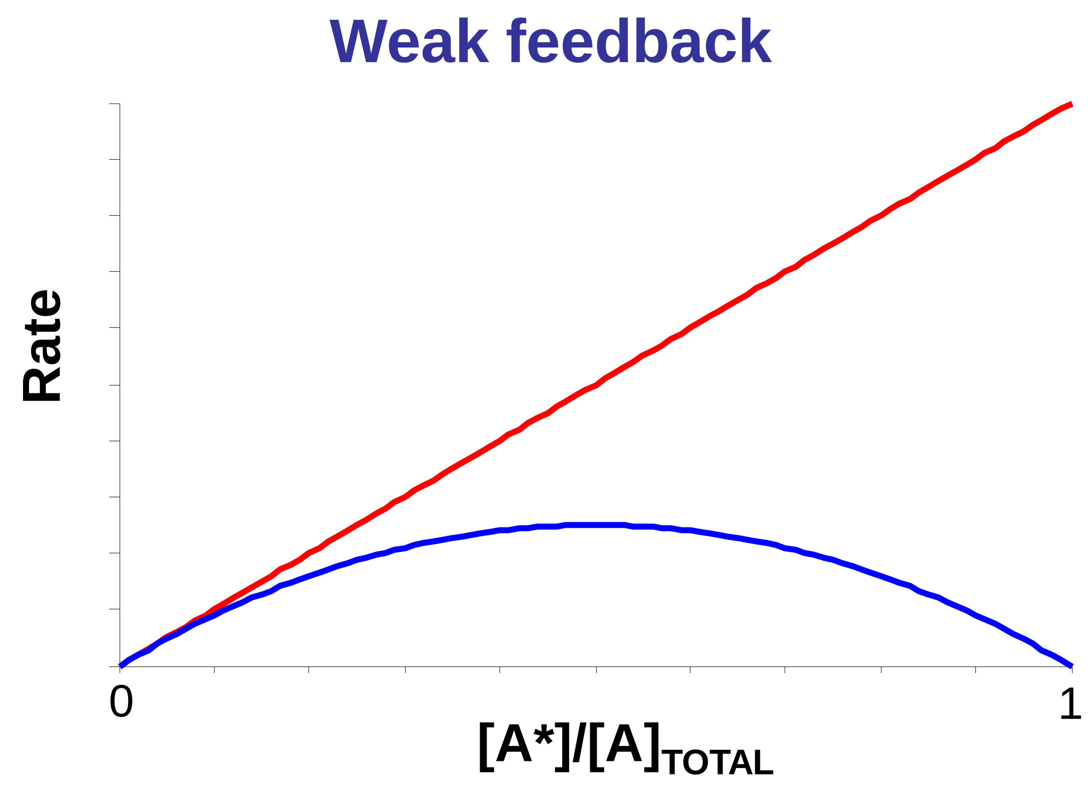
Rate balance plots

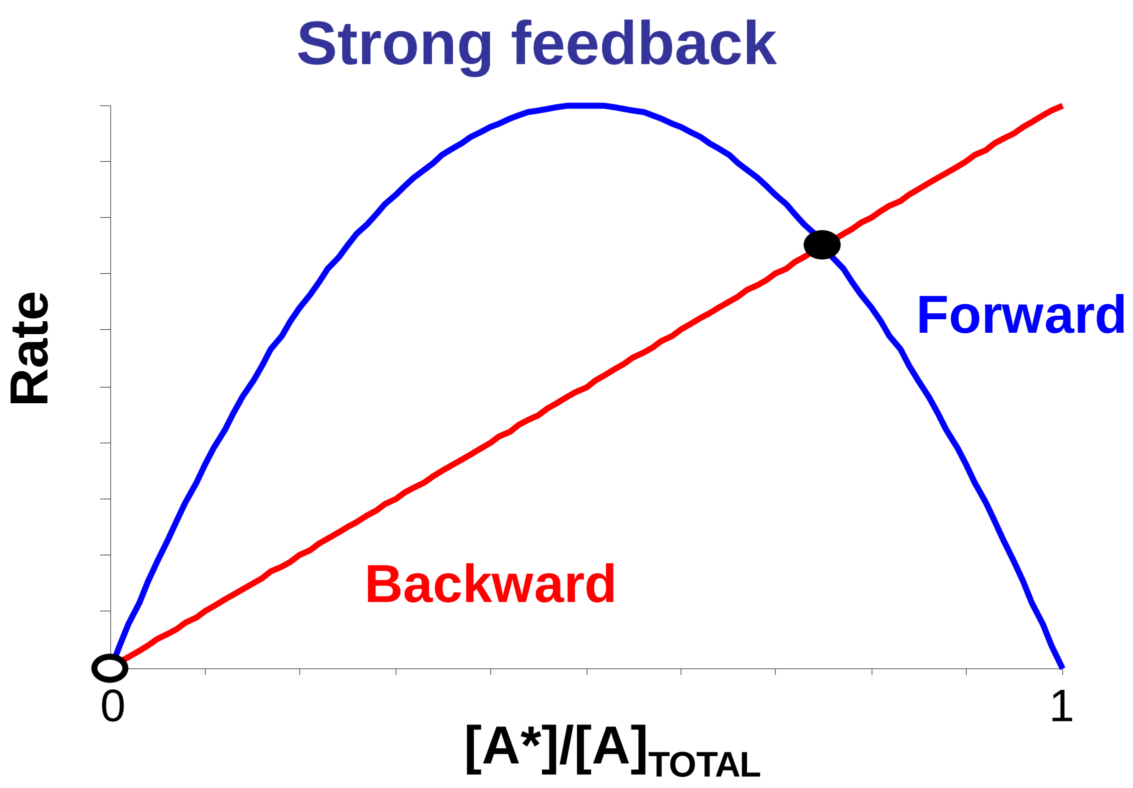
2) Michaelian system with linear feedback



$$FR = \left(k_+[S] + k_f[A^*]\right)\left([A]_{TOTAL} - [A^*]\right)$$

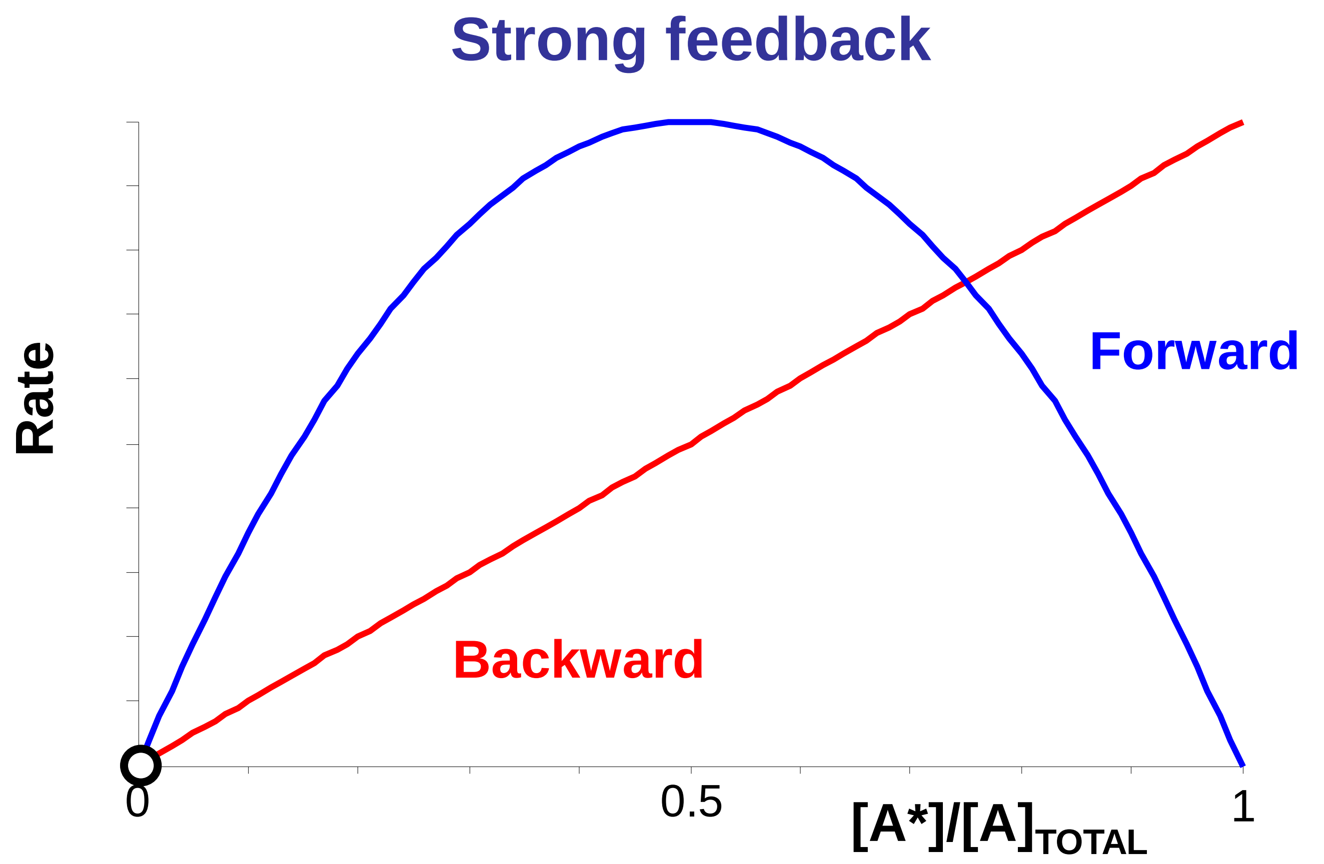
k_f determines strength of feedback





The right plot "looks" bistable. Is it? Answer: No.

Why is this plot not bistable?



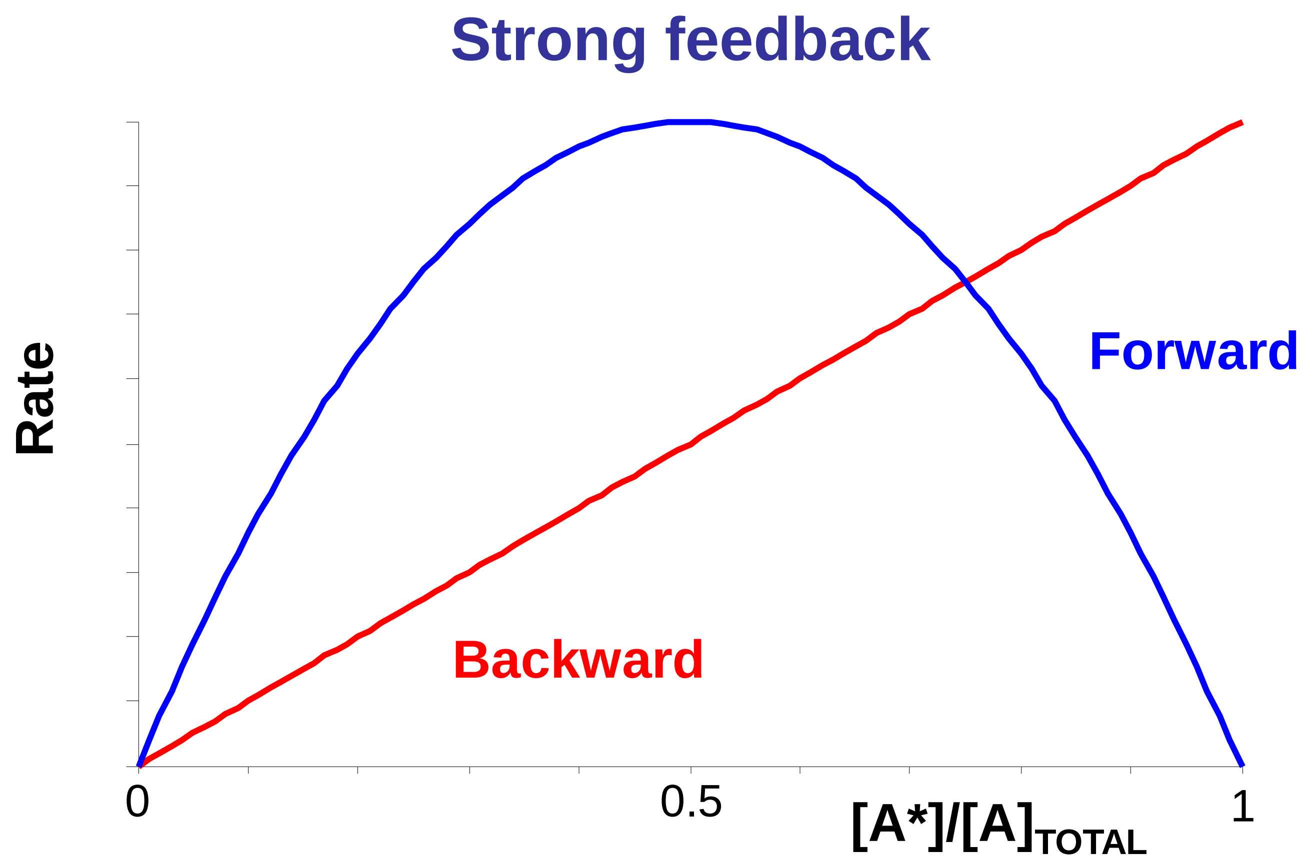
Consider a tiny deviation from A*=0, i.e. a spontaneous phosphorylation

Forward rate exceeds the backward rate.

This leads to a further increase in A*

Thus, this steady state is unstable

How can we make the "off" state stable?



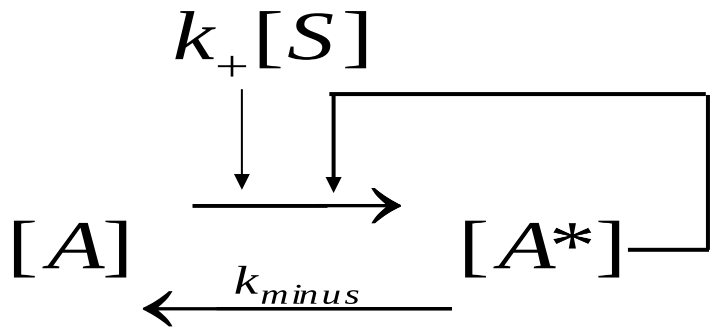
Two ways this can be modified to be truly bistable

1) Non-linear ("ultrasensitive") feedback

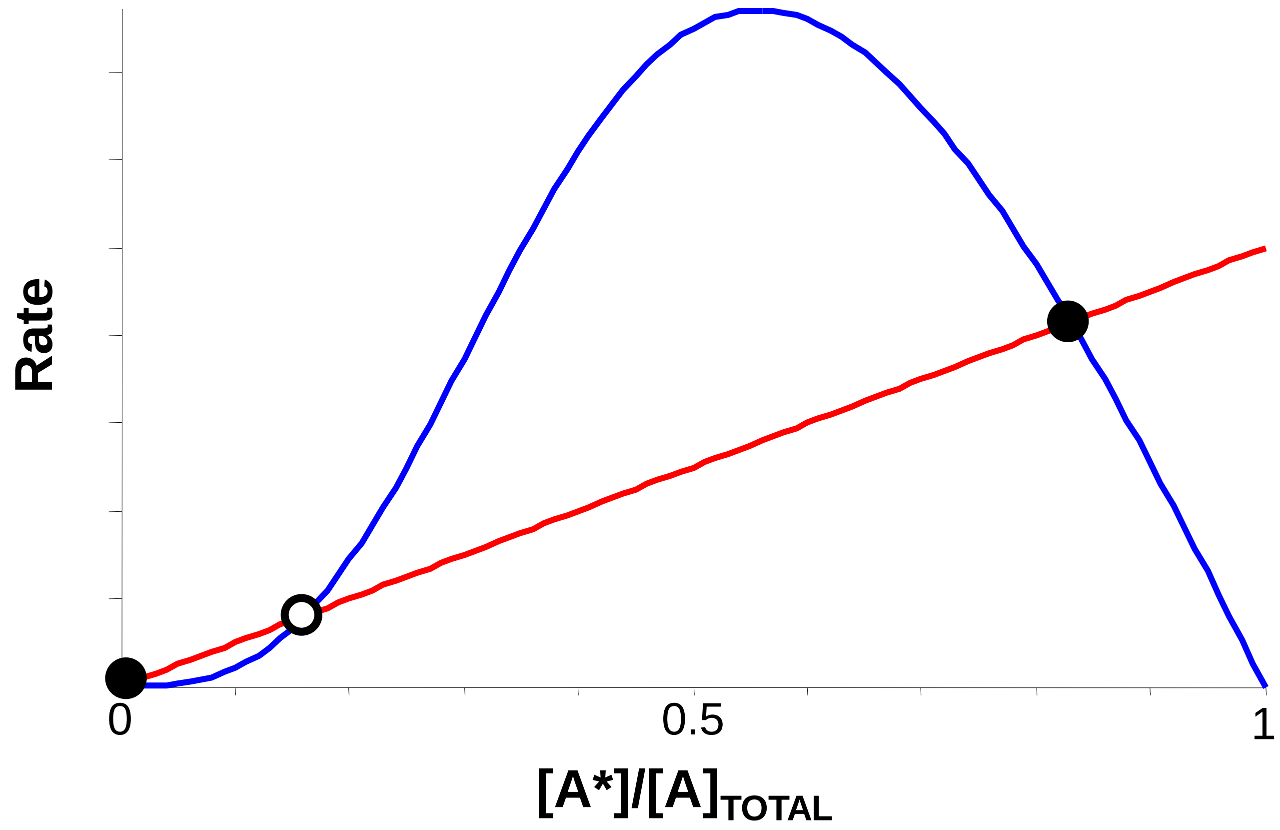
2) Partial saturation of the back reaction

Rate balance plots

3) Michaelian system with ultrasensitive feedback



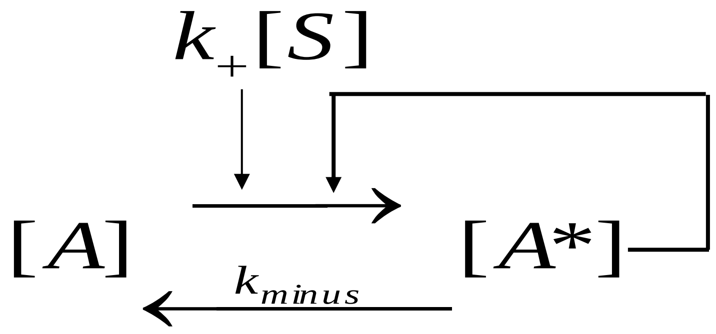
$$FR = \left(k_+[S] + k_f \frac{[A^*]^n}{[A^*]^n + K_{mf}^n} \right)([A]_{TOTAL} - [A^*])$$



Now we have a bona fide bistable system

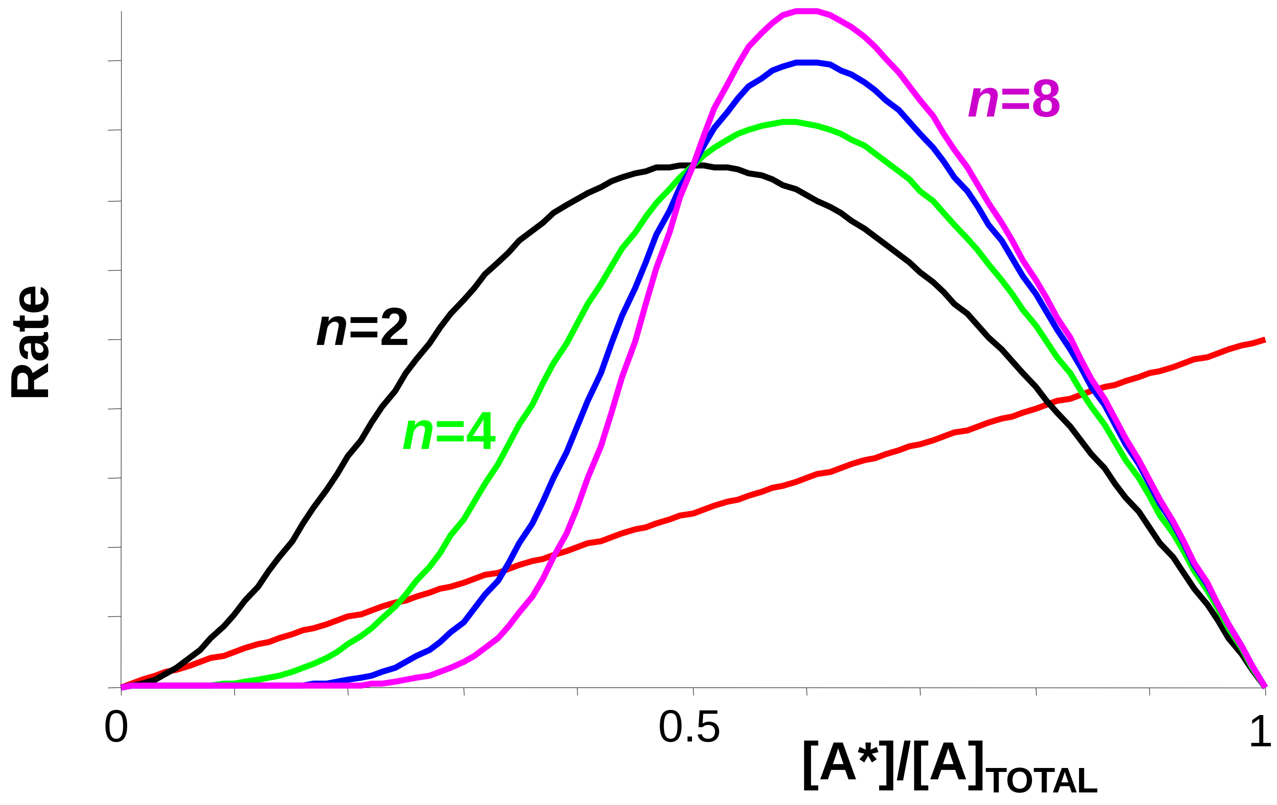
Rate balance plots

3) Michaelian system with ultrasensitive feedback



$$FR = \left(k_+[S] + k_f \frac{[A^*]^n}{[A^*]^n + K_{mf}^n} \right) \left([A]_{TOTAL} - [A^*] \right)$$

Effects of changes in hill exponent *n*



A larger hill exponent makes bistability more likely and more robust

Summary

Rate-balance plots are useful for assessing whether bistability may occur in one-variable systems.

Ultrasensitive positive feedback can produce bistability in a one-variable system.

Self-assessment question

You are working with an array `A`, dimensions 100 x 4. You also have a vector `time`, dimensions 100 x 1. Each column in `A` represents a different variable measured in your experiment. Each row represents the corresponding time point in the vector `time`. You wish to write a `for` loop to plot 4 time courses in different colors. You paste the following lines into your command window:

```
colors = ‘krgb’ ;
for i=1:4
  plot(time,A(i))
end
```

This does not produce the desired result for 3 reasons. Why not?

Answers: (1) You do not tell MATLAB to plot each column of `A`, instead it is only instructed to plot a single element of `A` each time through the loop.

(2) MATLAB has not been instructed to plot in a different color each time through the `for` loop.

(3) MATLAB has not been instructed to plot the time courses together.