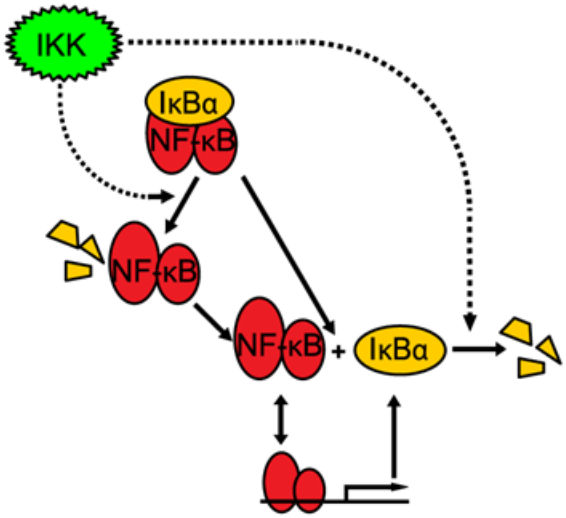
PhySci/MiMG/CaSB M178

Homework 6

Due: 11/08/22 at 12:00PM PDT

Problems

In class, we talked about NFκB signalling:



Longo et al. 2013, PLOS *Comput. Biol.* (https://doi.org/10.1371/journal.pcbi.1003112)

As a reminder, IκBα is a negative regulator of NFκB that is degraded with the activation of IKK, allowing NFκB to translocate into the nucleus and fulfill its role as a transcription factor. NFκB activation leads to downstream transcription of IκBα creating a feedback loop. In summary we have the following reactions:

Reactions	Description
$NF\kappa B + I\kappa B\alpha \xrightarrow{k\_a} NF\kappa B{:}I\kappa B\alpha$	Association of NFκB with IκBα
$NF\kappa B{:}I\kappa B\alpha \xrightarrow{k\_d} NF\kappa B + I\kappa B\alpha$	Dissociation of NFκB with IκBα
$IKK + NF\kappa B{:}I\kappa B\alpha \xrightarrow{r} IKK + NF\kappa B$	IKK-mediated degradation of IκBα bound to NFκB
$IKK + I\kappa B\alpha \xrightarrow{r} IKK$	IKK-mediated degradation of IκBα
$I\kappa B\alpha \xrightarrow{g}$	Constitutive degradation of IκBα
$NF\kappa B + prI\kappa B\alpha \xrightarrow{f\_a} NF\kappa B{:}prI\kappa B\alpha$	NFκB binding to IκBα promoter
$NF\kappa B{:}prI\kappa B\alpha \xrightarrow{f\_d} NF\kappa B + prI\kappa B\alpha$	NFκB unbinding to IκBα promoter
$prI\kappa B\alpha \xrightarrow{a} prI\kappa B\alpha + I\kappa B\alpha$	Constitutive synthesis of IκBα (delayed reaction)
$NF\kappa B{:}prI\kappa B\alpha \xrightarrow{b} NF\kappa B{:}prI\kappa B\alpha + I\kappa B\alpha$	Induced synthesis of IκBα (delayed reaction)

The synthesis of IκBα are "delayed reactions" because the cellular processes that compose protein synthesis (transcription, translation, nuclear import/export, etc.) take time. Hence in the model, the amount of IκBα produced at time t is dependent on the amount of active promoter present at time t-τ, where τ is the delay variable (i.e. the amount of time required for protein synthesis). We call such models "delay differential equations" (DDE). We can add the "_τ" ending to variable names to indicate that the reaction depends on the concentration of the species "τ" units of time in the past, thus we can rewrite our IκBα synthesis reactions as the following:

$prI\kappa B\alpha\_\tau \xrightarrow{a} prI\kappa B\alpha\_\tau + I\kappa B\alpha$	Constitutive synthesis of IκBα
$NF\kappa B{:}prI\kappa B\alpha\_\tau \xrightarrow{b} NF\kappa B{:}prI\kappa B\alpha\_\tau + I\kappa B\alpha$	Induced synthesis of IκBα

1. (20 points) First write down the change equations for the model described above. For the IκBα synthesis terms, write them in terms of the delayed variables, prIκBα_tau and NFκB:prIκBα_tau.

```
### implement the change equations
NFKB_prime = -k_a*NFKB*IKBa + k_d*NFKB_IKBa + r*IKK*NFKB_IKBa - f_a*NFKB*prIKBa +f_d*NFKB_prIKBa
IKBa_prime = -k_a*NFKB*IKBa + k_d*NFKB_IKBa - r*IKK*IKBa -g*IKBa + a*prIKBa_tau +b*NFKB_prIKBa_tau
NFKB_IKBa_prime = k_a*NFKB*IKBa - k_d*NFKB_IKBa -r*IKK*NFKB_IKBa
prIKBa_prime = -f_a*NFKB*prIKBa + f_d*NFKB_prIKBa
NFKB_prIKBa_prime =f_a*NFKB*prIKBa - f_d*NFKB_prIKBa
###
```

NFKB_prime = -k_a*NFKB*IKBa + k_d*NFKB_IKBa + r*IKK*NFKB_IKBa - f_a*NFKB*prIKBa +f_d*NFKB_prIKBa

IKBa_prime = -k_a*NFKB*IKBa + k_d*NFKB_IKBa - r*IKK*IKBa -g*IKBa + a*prIKBa_tau +b*NFKB_prIKBa_tau

NFKB_IKBa_prime = k_a*NFKB*IKBa - k_d*NFKB_IKBa -r*IKK*NFKB_IKBa

prIKBa_prime = -f_a*NFKB*prIKBa + f_d*NFKB_prIKBa

NFKB_prIKBa_prime =f_a*NFKB*prIKBa - f_d*NFKB_prIKBa

In the section of code called "NFKB - IKBa model" implement the change equations to simulate the model. Note that the values of prIκBα_tau and NFκB:prIκBα_tau are calculated within the model code in the section "delayed variables". You should use these values in your change equations implementation.

Once you have defined your model equations, run the cell containing the model code and the following cell titled "delay helper functions". These functions are used to look up the old values of prIκBα and NFκB:prIκBα needed by the model.

Before we can simulate NFKB responses to IKK activation, we have to find the steady state values of the model species to use as initial conditions for downstream

simulations. In the section, "Steady State Simulation", we will find the steady state values given some constraints. First, we assume there is a total of 100 nM of NFκB and "1 nM" of prIκBα (we approximate prIκBα as a continuous variable here, although in reality it can only take on nonnegative integer values). All other species are set to zero for now and this is implemented under "initial condition constraints". Below that we specify the model parameters. Note that all units are nM, $(nM*min)^{-1}$, or min^{-1}. Finally, we specify the parameters of IKK activation (IKK_on_time, IKK_off_time, IKK_amplitude), which are all set to zero to obtain the desired steady state result.

Run the section of code for "Steady State Simulation". The ODE solving looks a bit different from past homework, since here we need to solve the ODE system in pieces to call on past values of the delayed variables. **Because of this, the code will take a bit longer to run than in the past; be patient!** The last line of code in this section saves the final entry of the solution (steady_state) to be used as initial conditions for downstream simulations with IKK activation.

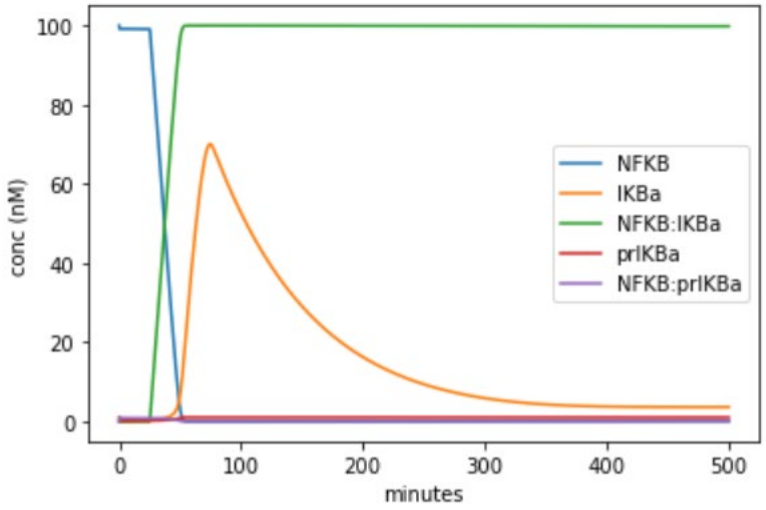
Run the section of code for "Checking Model Implementation". Check to see that the values on the right (from simulation of your model implementation) match the values on the left (from simulation of the correct model implementation). If the values don't match, double check your change equations and code before proceeding.

```
Checks on Model Implementation:
For each of the following lines ensure that your value on the right matches that on the left
48.225059139469735 48.225059139469735
96.78765543513467 96.78765543513467
52.60671701101044 52.60671701101044
16.23230531035957 16.23230531035957
0.9981694760165793 0.9981694760165793
```

They match!

Run the section of code called "Plot Dynamics" and paste your graph here. State whether your solution appears to reach steady state. If so, we can use the saved steady state value. If not, increase the value of the variable *iterations* (default = 20) in the steady state simulation section. Rerun the steady state simulation and plotting until satisfied with the steady state result.

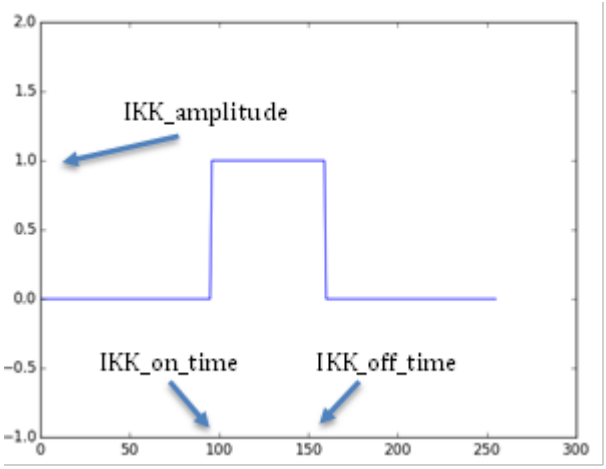
My solution appears to reach steady state: NFKB is depleted to 0 after 20 iterations, IKBa is also depleted to 0 because of the full concentration of the NFKB:IKBa complex being made. PrIKBa and NFKB:prIKBa concentration stayed at 0 throughout.



What's the dominant form of NFκB at steady state (free or bound to IκBα)?

The dominant form of NFKB at steady state is bound to IKBalpha, as shown in the graph.

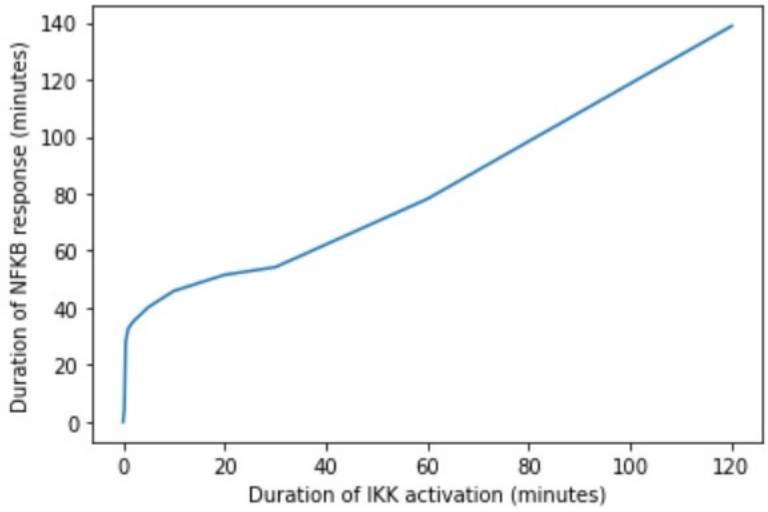
2. (20 points) Now we will simulate NFκB responses to IKK activation. In these simulations, the profile of IKK activation will look like a box function:



Run the section of code "NFKB responses to IKK activation" to define our model simulations. We are just putting the code that solves for our model inside of a function call so we can more easily reuse this piece of code. **Next, we will run the section of code called "Duration of NFKB response versus IKK duration".** For these simulations we will activate IKK at time = 0 (IKK_on_time = 0) and set its amplitude

to 10, a high value. We will then vary the duration of IKK activation by changing IKK_off_time, all the way from 15 second (0.25 minutes) to 2 hours (120 minutes). For each IKK activation profile, the simulation will run and the code then finds the amount of time the NFκB response is above a specified threshold (NFKB_threshold = 1 by default).

Run the section of code "Plot IKK activation versus NFKB response duration" and paste the resulting graph here.

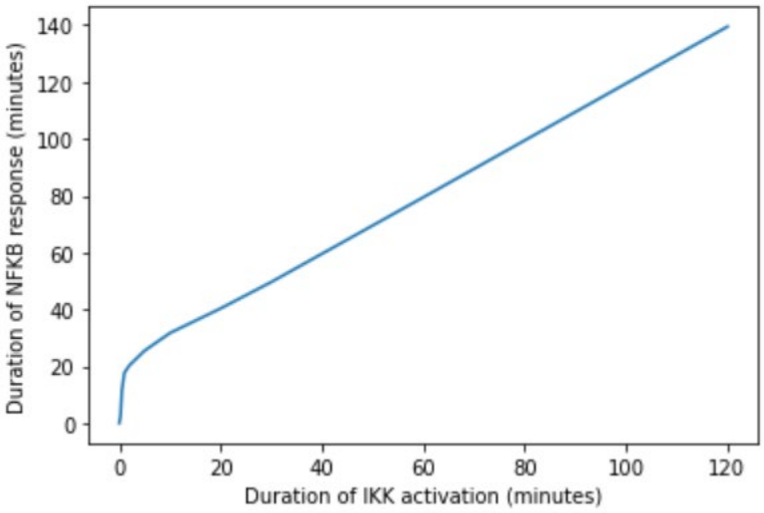


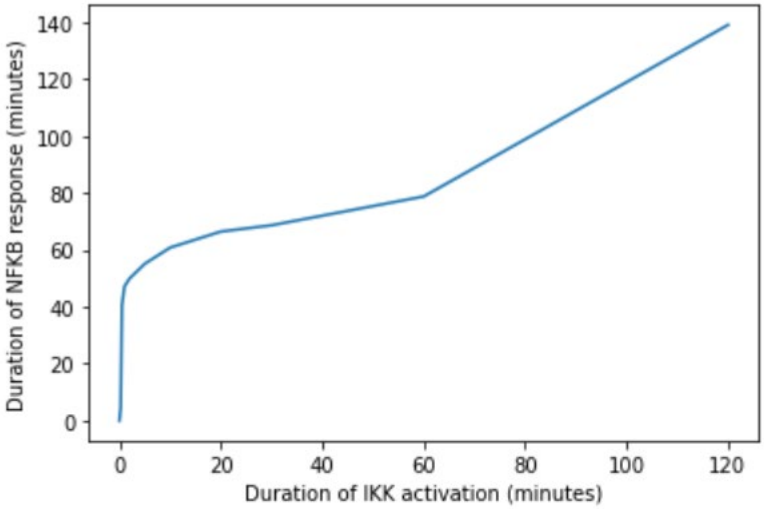
How does NFKB response duration vary with the duration of IKK activation? Can you mechanistically explain the difference in behavior for low values of IKK activation duration versus high values?

The linear trajectory is only persistent after around 35 minutes of IKK activation, prior of which looks like a logarithmic increase trend. The difference in behavior for low values of IKK is that when there are higher duration of IKK activation, it means that there were more inflammatory signals. There is IKBa negative feedback which turns off NFKB, so that at higher values the response will not be at an increasing rate as it was in the lower values.

Now we will rerun this entire exercise with a different value for the delay parameter. We will try the delay parameter, tau, set to 10 minutes and 40 minutes. Return to the section of code called "Steady State Simulation". Change the value of tau accordingly and rerun the section of code. Next rerun the section of code "Plot Dynamics" and once again ensure steady state is reached. Then rerun the section of code "NFKB responses to IKK activation" and "Duration of NFKB response versus IKK duration".

Finally, plot the result from the section “Plot IKK activation versus NFKB response duration”. Paste your resulting graph here and repeat the entire process for the other value of tau.





How does the relationship between duration of IKK activation and NFKB response duration change with different values for the delay parameter, tau?

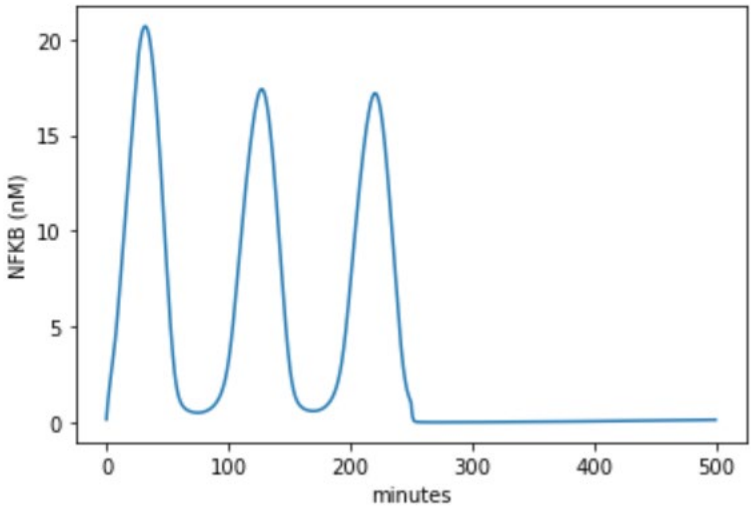
Without delay, we see a NFKB response right away at around 3 minutes or 5 minutes of duration of IKK activation when the time delay parameter is 10 (units); With delay, the duration of IKK activation becomes linear around 60 minutes. This is because, with a delay, we might see the expression of IKBe as a fail-safe mechanism. At a time delay parameter of 40 (units), we see that the duration of NFKB response is longer

until linearity at around 80 minutes as opposed to 20 minutes in the precedent example.

3. (20 points) Finally we will simulate NFκB responses to changing amplitudes of IKK activation. For these simulations, we will maintain a constant duration of IKK activation, 250 minutes (IKK_on_time = 0, IKK_off_time = 250). In the section titled "NFKB response versus amplitude of IKK activation (1)" you can vary the amplitude of IKK activation (final argument of function call NFKB_response, default value = 1).

Before running this section of code, reset the delay parameter, tau, to 25 minutes. Rerun the "Steady State Simulation" and "NFKB responses to IKK activation" sections so the change takes effect.

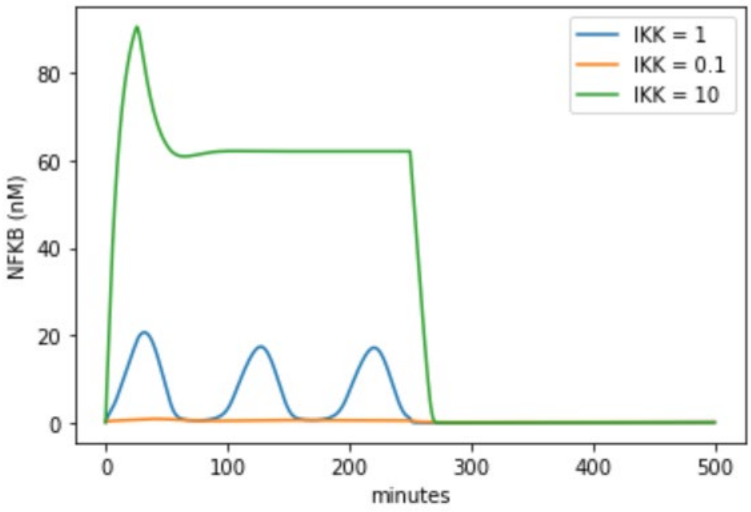
Now run the code with the default amplitude of 1 and paste the resulting graph. Describe the activation dynamics of NFκB.



When IKK activating NFKB is not transient and when the concentration of NFKB is not too high, also considering the condition that there is delay (tau), there will be oscillations due to delayed negative feedback. This is when the amplitude is low, such as 1. We see the oscillation behavior happening for about 250 minutes, which is a very long time in this system.

Run the section of code "NFKB response versus amplitude of IKK activation (2)" and paste the resulting graph. This plot includes NFκB responses to IKK amplitudes of

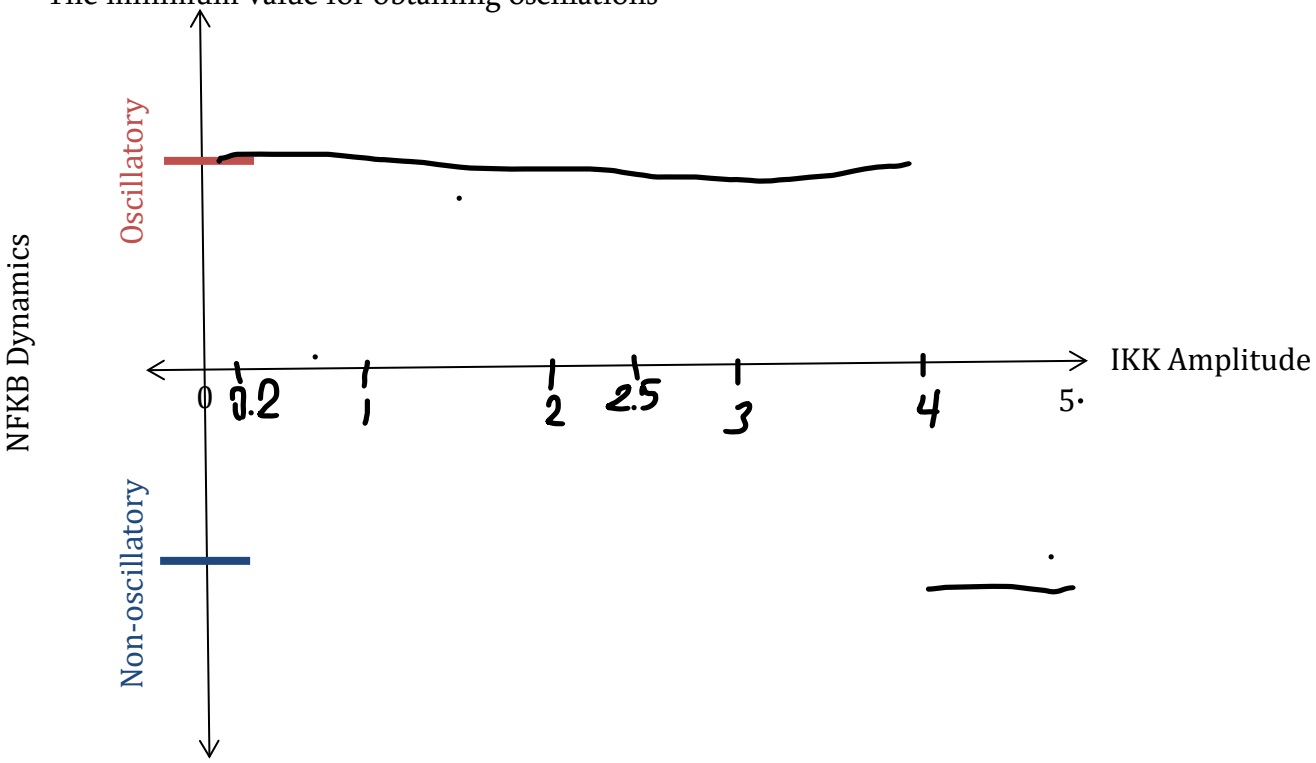
0.1, 1, and 10 on the same plot. How might you classify these three types of responses?

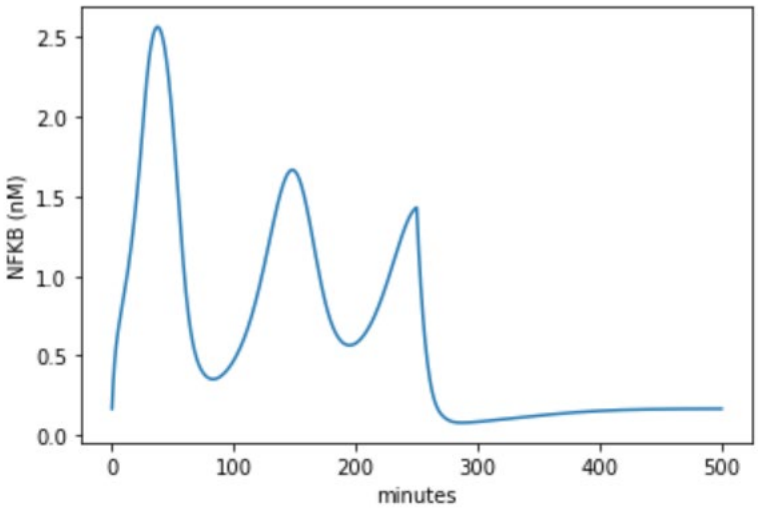


The IKK = 0.1 response is no activation of NFKB because the amplitude is too low for oscillation behavior. The IKK = 1 is an example of delayed negative feedback, providing an oscillation behavior with amplitudes of the oscillations around 20 nM NFKB activation. The IKK = 10 shows the nature of IKBa not being able to maintain that oscillation behavior at too high of an amplitude, when IKK remains high.

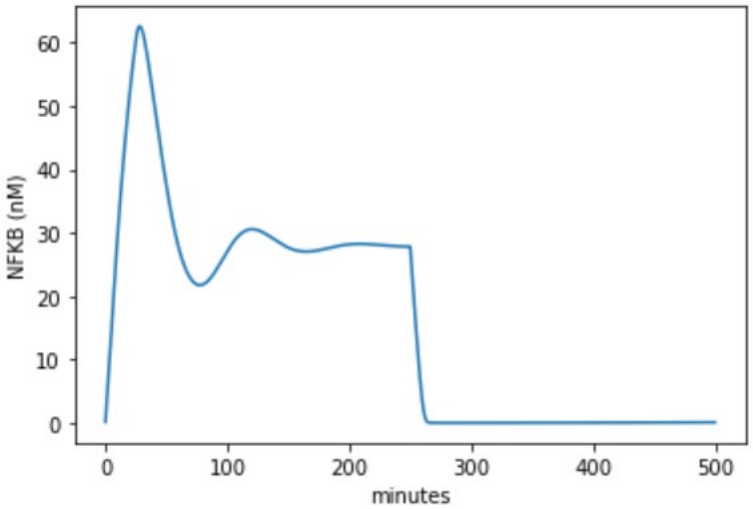
Rerun the section of code "NFKB response versus amplitude of IKK activation (1)" with different values of IKK amplitude. Find the minimum value needed to obtain oscillations (to the nearest tenth) and then the minimum value beyond that needed to eliminate the oscillations (to the nearest integer). **Draw your results onto the graph below** where the x-axis is IKK amplitude (from 0 to 5) and the y-axis is NFKB response behavior (let 1 = oscillatory, -1 = non-oscillatory).

The minimum value for obtaining oscillations





For an amplitude of 0.2, the "IKK off" time of 250 minutes shows the last viable oscillatory behavior. I would say 0.2 is the minimum for maintaining the oscillatory behavior.



This is the graph of an amplitude of 4, and we can see that the oscillatory behavior doesn't propagate to 250 minutes. We can safely estimate that the closest integer value is 4, because 3 we still notice the oscillatory behavior (albeit diminishing).