

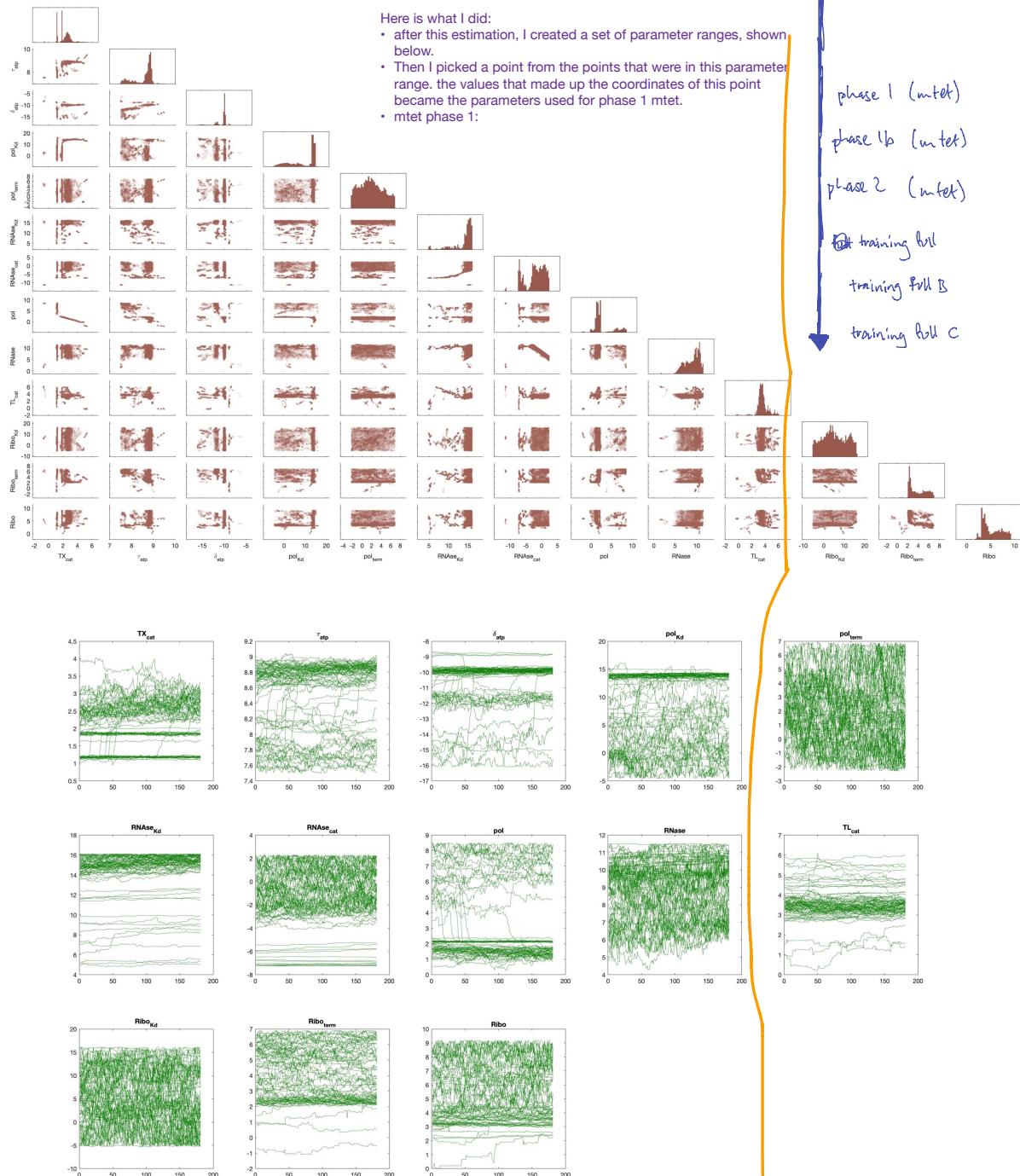
Overall, we can only effectively optimize up to 15 params. so we need to take a stepwise approach.

VNPRL

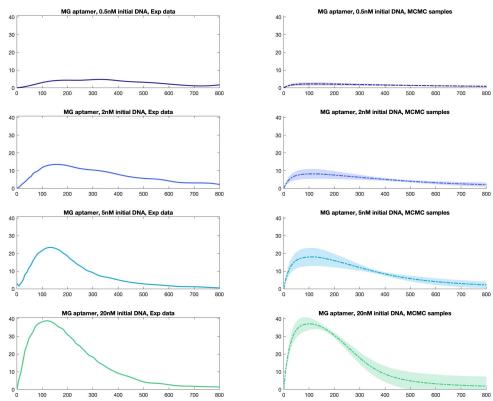
simdata_20190223_024333_1_2

/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_vnprl/

MARL (F2)



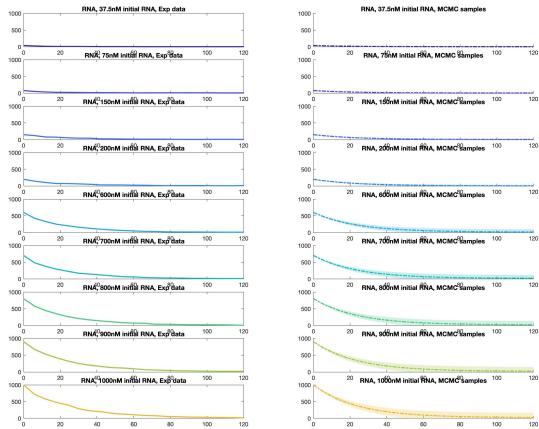
simdata_20190223_024333_1_2/cornerplot20190223_024333_1_2BurnedIn.jpg



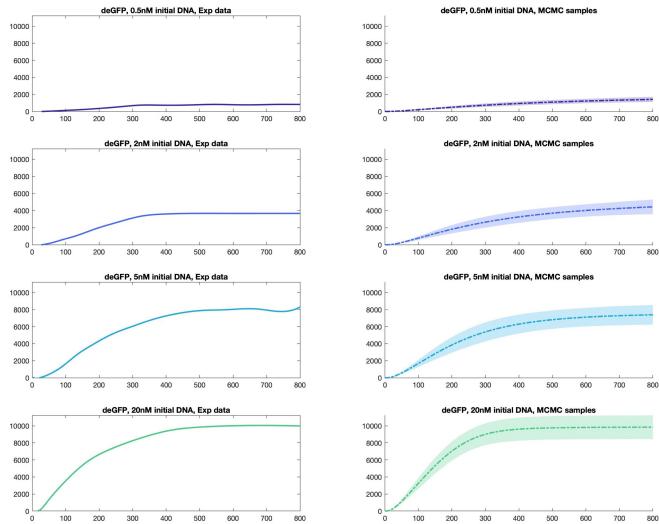
vnprl_F2

```
% FINALLY USED VALUES SECTION %%%%%%
% Ok, next up, we are going to restrict most parameters to some ranges, and
% then pick parameter values at random from the set of remaining points.
% TX cat: 2.3 2.95
% RNase_Kd (index 6) ----> 15.59 15.73
% RNase_cat (index 7) ----> -0.2966 0.1031
% RNase (index 9) ----> 8.385 8.622
close all
paramIndices = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13];
parRanges(paramIndices, :) = [...]
    2.3 2.95; %1
    8.6 8.95; %2
    -10.1 -9.7; %3 delta atp (deg rate)
    13 15; %4 pol Kd tet
    -1 5 ; %5 pol term
    15.59 15.73; %6 rnase KD
    -0.2966 0.1031; %7rnase cat
    0.7 2.2; %8 pol
    8.385 8.622;%9 RNase
    3.18 3.69; % 10 -- TLCat
    -3 13.5;%RiboKd
    2 3;% Ribo term
    3 4.6 ];% Ribo
marray_cut = mcmc_cut(marray, paramIndices, flipud((parRanges(paramIndices, :))));
mcmc_plot(marray_cut(:, 1:end,ceil(end/4):end), parnames(), ...
'saveasmatlabfig', figsave, 'savejpeg', false, ...
'projdir', projdir, 'tstamp', tsToSave, 'extrafignamestring', 'BurnedIn');

CandidateParams = marray_cut(:, 1:100:end, end)
```



```
parRanges(paramIndices, :) = [...]
    2.3 2.95; %1 tx cat
    8.6 8.95; %2 tau
    -10.1 -9.7; %3 delta atp (deg rate)
    13 15; %4 pol Kd tet
    -1 5 ; %5 pol term
    15.59 15.73; %6 rnase KD
    -0.2966 0.1031; %7rnase cat
    0.7 2.2; %8 pol
    8.385 8.622;%9 RNase
    3.18 3.69; % 10 -- TLCat
    -3 13.5;%RiboKd
    2 3;% Ribo term
    3 4.6 ];% Ribo
```



/Users/vipulsinghal/Dropbox/
Documents/toolbox/
txtlsim_vsfork2017/mcmc_simbio/
projects/proj_vnprl/
simdata_20190223_024333_1_2/
traj20190218_135635_1_242MGa_de
GFP.jpg

[82ef659805d5d402b6644706528be893ccd8e964](#)

phase 1, mtet: (just ptet, ptet + tetR, ptet+tetR+atc)

```

+
% Thus, the 7 estimated parameters are:
3 % 'TXTL_PTET_RNAPbound_Kd' , 'est'
4 % 'TXTL_PTET_sequestration_Kd' , 'est'
5 % 'TXTL_PTET_sequestration_F' , 'est'
6 % 'TXTL_INDUCER_TETR_ATC_Kd' , 'est'
7 % 'TXTL_INDUCER_TETR_ATC_F' , 'est'
8 % 'TXTL_DIMER_tetR_Kd' , 'est'
9 % 'TXTL_DIMER_tetR_F' , 'est'

+
% preFixedParams = ...
10 'AGTReg_ON' , exp( -3.9120)
11 'TXTL_PROT_deGFP_MATURATION' , exp( -6.0748)
12 'TXTL_PTET_RNAPbound_F' , exp( 1.5000)
13 'TXTL_NTP_RNAP_1_Kd' , exp( 2.9459)
14 'TXTL_NTP_RNAP_2_Kd' , exp( 13.9970)
15 'TL_AA_Kd' , exp( 6.5566)
16 'TL_AGP_Kd' , exp( 14.5090)
17 'TXTL_UTR_UTR1_F' , exp( -0.2000)
18 'TXTL_PLAC_RNAPbound_F' , exp( 1.5000)
19 'TXTL_NTP_RNAP_1_F' , exp( 0)
20 'TXTL_NTP_RNAP_2_F' , exp( 0)
21 'TL_AA_F' , exp( -0.3000)
22 'TL_AGP_F' , exp( -1.2000)
23 'TXTL_RNAdeg_F' , exp( 0); %

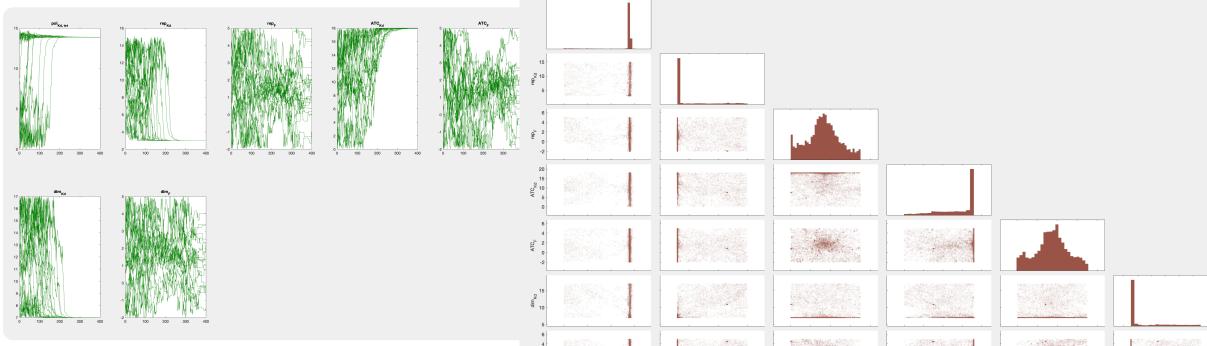
+
parnames = ...
[{'pol_(Kd, tet)'}
 {'rep_(Kd)'}
 {'rep_F'}
 {'ATC_(Kd)'}
 {'ATC_F'}
 {'dim_(Kd)'}
 {'dim_F'}];

```

%

```

% tstamp = [t11 t12 t13 t14 t15 t16 t17 t18 t19 t10 t11];
% ntstampID = [1:2 1:10 1:8 1:5 1:5 1:5 1:5 1:5 1:5 1:5 1:5];
% tstamp = [t11 t12 t13 t14 t15 t16 t17 t18 t19 t10 t11];
% ntstampID = [1:2 1:10 1:8 1:5 1:5 1:5 1:5 1:5 1:5 1:5];
load([projdir '/models/_tst1 /full_varable_set_.mat']);
[m1,...]
'mmc_info', 'data_info', 'm1', 'r1');
conclusions from case 2:
```



```

{'TXTL_PTET_RNAPbound_Kd' } {[ 0] } {[17]}
{'TXTL_PTET_sequestration_Kd' } {[ 3] } {[15]}
{'TXTL_PTET_sequestration_F' } {[ -2]} {[ 5]}
{'TXTL_INDUCER_TETR_ATC_Kd' } {[ 0] } {[18]}
{'TXTL_INDUCER_TETR_ATC_F' } {[ -2]} {[ 5]}
{'TXTL_DIMER_tetR_Kd' } {[ 7] } {[17]}
{'TXTL_DIMER_tetR_F' } {[ -2]} {[ 5]} ----- these were the parameter ranges for phase 1 estimated params. we can see from the graph above that

```

rep kd, atc kd and dim kd are hitting the boundaries, and need to be reestimated. we fix the rest, since they are either F rates or really tightly estimated kds:

```

preFixedParams2 = {
7      'TXTL_PTET_RNAPbound_Kd' , exp(14)
12     'TXTL_PTET_sequestration_F' , exp(1.314)
17     'TXTL_INDUCER_TETR_ATC_F' , exp(1.577)
19     'TXTL_DIMER_tetR_F' , exp(1.447)};

```

These were found in phase 1, and fixed for phase 1b.

```

mtet_phase1_params = ...
{...
7      , 'TXTL_PTET_RNAPbound_Kd' , exp(14)
11     , 'TXTL_PTET_sequestration_Kd' , exp(0) , [exp(0) exp(25)]
12     , 'TXTL_PTET_sequestration_F' , exp(-1) , [exp(-10) exp(5)]
12     , 'TXTL_PTET_sequestration_F' , exp(1.314) , [exp(-2) exp(5)]
}

```

```

16      ,      'TXTL_INDUCER_TETR_ATC_Kd'      ,      exp(-2)      ,      [exp(-15)      exp(5)]
17      ,      'TXTL_INDUCER_TETR_ATC_F'       ,      exp(1.577)    ,      [exp(-2)      exp(5)]
18      ,      'TXTL_DIMER_tetR_Kd'        ,      exp(-10)      ,      [exp(-20)      exp(-7)]
19      ,      'TXTL_DIMER_tetR_F'         ,      exp(1.447)    ,      [exp(-2)      exp(5)]...
};

The three highlighted parameters were the ones estimated in phase 1b, and those were the values found.

```

```

% 13x1 cell array
% ('TXL(cat)' ) 1
% ('tRNA(cat)' ) 3
% ('tRNA(atp)' ) 3
% ('sdeRNA(gp1)' ) 5
% ('pol_(Kd)' ) 21 % plac Kd NOT PTET Kd, the PtET Kd gets estimated.
% ('pol_(tern)' ) 23
% ('pol_(tetR)' ) 25
% ('tRNA_(cat)' ) 18
% ('pol' ) 31
% ('tRNA(atp)' ) 32
% ('sdeRNA(gp1)' ) 27
% ('Ribo_(Kd)' ) 6
% ('Ribo_(tern)' ) 28
% ('Ribo_(tetR)' ) 33
%
% 13x1 cell array
EstimatedParams =:, % 
- 2.5234 2.4664 ('TXL(cat)' ) 1
8.8854 8.9024 ('tRNA(atp)' ) 3
-10.0179 -9.2713 ('sdeRNA(gp1)' ) 5
1.1711 13.6101 ('pol_(Kd)' ) 21 % plac Kd NOT PTET Kd, the PtET Kd gets estimated.
2.0800 4.3598 ('pol_(tern)' ) 23
15.6977 15.6349 ('tRNAse_(Kd)' ) 15
8.0525 -8.4462 ('tRNAse_(cat)' ) 15
8.0525 8.1457 ('pol' ) 31
8.3976 8.6141 ('tRNA(atp)' ) 32
3.6442 3.2731 ('sdeRNA(gp1)' ) 2
8.2565 8.0542 ('Ribo_(Kd)' ) 3
2.5533 2.6328 ('Ribo_(tern)' ) 28
3.9983 4.3863 ('Ribo_(tetR)' ) 33
%
EstimatedParams =:, % 
ParamColumnTolse = 2
paramVecTolse = Est1;
indiceMasterVectorDf = -10.0179 -9.2713 -0.9376 -10.0048 -9.9856 -9.8889 -9.9100 -9.9631 -9.9827 -10.0348 -9.9782 -9.9538
+ 13.7327 13.6474 13.8789 13.9325 13.7827 13.7665 13.7762 13.8681 13.8867 13.7211 14.0961 13.8786
+ 2.5234 2.4664 2.6231 2.5976 2.4658 2.4277 2.4886 2.6018 2.4381 2.5499 2.6991 2.4356
+ 8.8854 8.9024 8.0807 8.8014 8.8483 8.8642 8.8394 8.8427 8.8616 8.7634 8.8621 8.7918
+ 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525 0.0525
+ 0.0525 -0.2251 -0.1516 -0.8789 -0.2109 -0.1276 -0.0340 -0.1752 -0.8117 0.0436 -0.2481 -0.1683
+ 1.5652 1.6167 1.4897 1.5286 1.6088 1.6259 1.6088 1.5196 1.7133 1.5597 1.4531 1.7096
+ 8.2565 8.6141 8.6141 8.5125 8.5118 8.5146 8.4965 8.5494 8.4957 8.4027 8.6184 8.5097
+ 3.6442 3.2731 3.2731 3.2658 3.2658 3.2658 3.1941 3.2087 3.1474 3.4028 3.3689 3.3803
+ 2.5533 2.6328 2.8292 2.7929 2.9282 2.8884 2.9859 2.9519 2.8887 2.7658 2.7136 2.9940
+ 3.9983 4.1863 4.1261 4.0812 4.1922 3.9954 4.0318 3.7338 3.8616 4.1838 3.8526 3.8881];
%
ParamColumnTolse = 2;
paramVecTolse = EstimatedParams(:, ParamColumnTolse);
+ indiceMasterVectorEstimated = [1 3 5 21 23 15 30 31 32 2 6 28 33];

```

/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet/
simdata_20190420_155505_8_31/cornerplot20190420_155505_8_31AllWalkers.fig
/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet/
simdata_20190420_155505_8_31/trace20190420_155505_8_31AllWalkers.fig

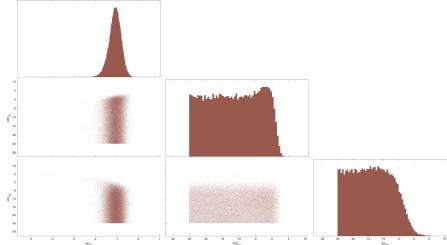
mtet phase 1b is different from phase 1. the parameters estimated are

```
parnames = ...
[{'rep_{Kd}'}
 {'ATC_{Kd}'}
 {'dim_{Kd}'}
];
```

```
case 3
ts1 = '20190421_155749_1_773';
ts2 = '20190421_155749_2_309';
ts3 = '20190422_142534_1_773';
ts4 = '20190422_214228_1_773';
ts5 = '20190424_104801_1_773';

tstamp = {ts1 ts2 ts3 ts4 ts5};
niterID = (1:10 1:2 1:4 1:7 1:14);
load(['projdir' '/simdata' ts1 '/full_variable_set_ ts1 '.mat'], ...
'mi',...
'mcmc_info',...
'data_info',...
'mai',...
'ri');

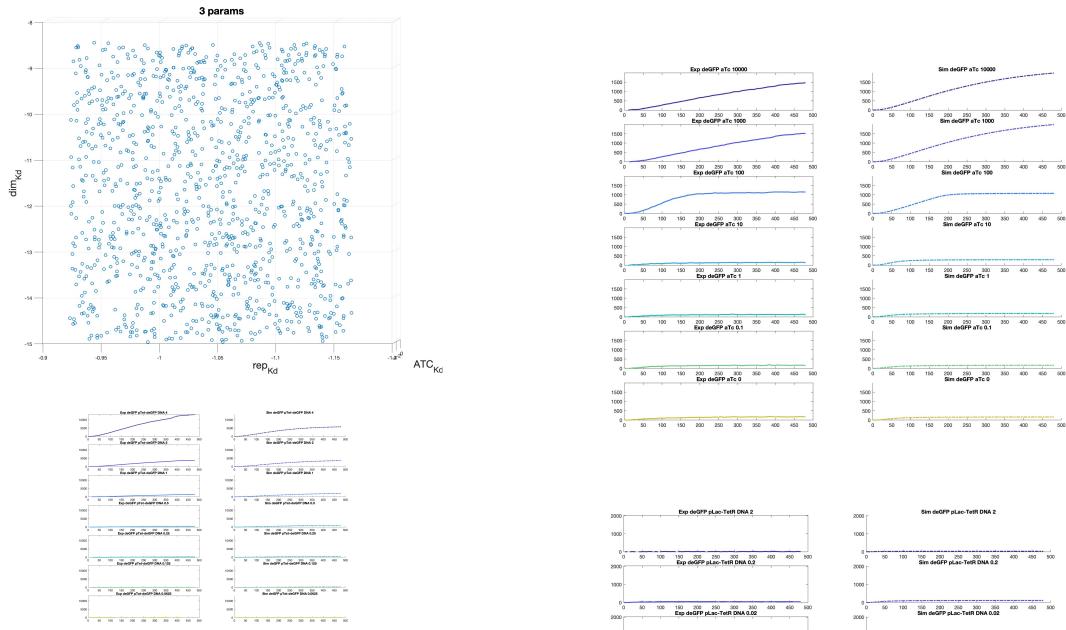
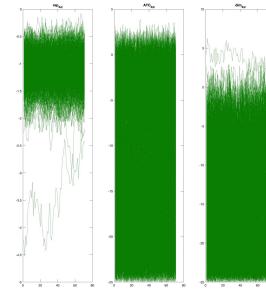
end
tsToSave = ts5;
mai.masterVector
```



- 1, -2, -10 are the parameter values found here. see above.

the files are sadly in the same folder: /Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet

ae29a07bb7960993c54f5d4f96a7ca395e446f5d



/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet/simdata_20190424_104801_1_773

SCREEN RECORDING OF CARTESIAN PRODUCT.

/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet/simdata_20190424_104801_1_773
No_cov_fig_cool.mov

phase 2: add the plac constitutive production to the mix, and different set of parameters.

```
mcmc_info_ZTFI_mix_phase2.m
% parameters of the master vector we fix and estimate
%
% The following params are the ones estimated from the last set of
% restrictions in the file analysis_vnprl_F2.m
%
% parameters
%
% 1x1 cell array
%
% {'Tx_(cat)' } 1
% {'TxL_(cat)' } 2
% {'Tdelta_(cat)' } 5
%
% {'pol_(Kd)' } 21 % plac Kd NOT PTET Kd, the PtET Kd gets estimated.
%
% {'RNase_(cat)' } 1
% {'RNase_(Kd)' } 15
% {'RNase_(cat1)' } 30
% {'RNase' } 31
% {'RNase' } 32
% {'TL_(cat)' } 2
% {'RIBO_(Kd)' } 6
% {'RIBO_(terms)' } 28
% {'Ribo' } 33
```

% The following were parameters estimated in vnprl_F2, and were extracted
% to a master vector. These are the ones we need to fix to
% understand how they were picked. Need to document this before I forget.
EstimatedParams = ...
2.6331 2.5916 2.4631 2.4277 2.4086 2.4819 2.4301 2.5499 2.6991 2.4356
8.9824 8.8997 8.8914 8.8483 8.8942 8.8394 8.8427 8.7634 8.8621 8.7918
-10.0179 -9.7873 -9.9377 -10.8044 -9.9056 -9.8889 -9.5108 -9.9631 -9.9827 -10.8348 -9.8762 -9.9538
13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323 13.7323
2.0880 4.3598 8.8928 2.8127 3.5498 4.4985 4.1178 3.1338 8.3334 2.7831 1.8745 1.7856
15.6977 15.6349 15.6893 15.6661 15.6339 15.7183 15.6193 15.6295 15.7294 15.6674 15.5958
0.4162 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562 1.0562
1.5652 1.6167 1.4897 1.5296 1.6088 1.6259 1.6088 1.5196 1.7133 1.5597 1.4531 1.7896
8.3976 8.6141 8.6149 8.5123 8.5178 8.5146 8.4985 8.5494 8.3963 8.4382 8.6184 8.5897
3.0442 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242 2.4242
0.2565 0.0542 4.5196 -2.2731 -1.6399 -1.7370 -2.9377 0.6546 -1.7598 3.9814 6.7136 -0.3827
2.5533 2.8328 2.8292 2.7929 2.9282 2.8884 2.9559 2.9519 2.8887 2.7650 2.7138 2.9940
3.5983 4.1863 4.1261 4.0012 4.1922 3.9954 4.8318 3.7338 3.9816 4.1088 3.8526 3.8881;

```
ParamColumnTolUse = 2;
paramColUse = EstimatedParams(:, ParamColumnTolUse);
IndicesMasterVectorEstimated = [1 3 21 23 15 38 31 32 2 6 28 33];
```

- parameter ranges the estimation in phase 2 was carried out over.
- { 'TX_elong_glob' } { [0] } { [5] }
- { 'TL_elong_glob' } { [0] } { [6] }
- { 'AGTPdeg_time' } { [6] } { [18] }
- { 'AGTPdeg_ON' } { [0] } { [25] }
- { 'AGTPdeg_Rate' } { [-10] } { [5] }
- { 'TXTL_UTR_UTR1_Kd' } { [10] } { [5] }
- { 'TXTL_PTET_RNAPbound_Kd' } { [14] } { [17] }
- { 'TXTL_PTET_RNAPbound_F' } { [1.5] } { [4] }
- { 'TXTL_INDUCER_TETR_ATC_Kd' } { [2.9] } { [5] }
- { 'TXTL_NTP_RNAP_2_Kd' } { [14] } { [20] }
- { 'TXTL_PTET_sequestration_Kd' } { [12] } { [15] }
- { 'TXTL_PTET_sequestration_F' } { [1.5] } { [5] }
- { 'TL_AA_Kd' } { [6.6] } { [3] }
- { 'TL_AGT_P_Kd' } { [14.5] } { [10] }
- { 'TXTL_RNAdeg_Kd' } { [15.2] } { [7] }
- { 'TXTL_INDUCER_TETR_ATC_Kd' } { [13] } { [18] }
- { 'TXTL_INDUCER_TETR_ATC_F' } { [2.6] } { [5] }
- { 'TXTL_DIMER_tetR_Kd' } { [13] } { [17] }
- { 'TXTL_DIMER_tetR_F' } { [2.6] } { [5] }
- { 'TXTL_UTR_UTR1_F' } { [1.2] } { [4] }
- { 'TXTL_PTET_RNAPbound_Kd' } { [13.8] } { [5] }
- { 'TXTL_PLAC_RNAPbound_F' } { [2.6] } { [5] }
- { 'TXTL_RNAPBOUND_TERMINATION_RATE' } { [1.8] } { [3] }
- { 'TXTL_NTP_RNAP_1_F' } { [0] } { [2] }
- { 'TXTL_NTP_RNAP_2_F' } { [0] } { [3] }
- { 'TL_AA_F' } { [0.3] } { [3] }
- { 'TL_AGT_F' } { [-1.2] } { [4] }
- { 'TXTL_RIBOBOUND_TERMINATION_RATE' } { [0] } { [5] }
- { 'TXTL_RNAdeg_F' } { [0] } { [3] }
- { 'RNase' } { [0.45] } { [-5] }
- { 'RNase' } { [1.4419] } { [1] }
- { 'Ribo' } { [8.5] } { [12] }
- { 'Ribo' } { [3.75] } { [12] }
- { 'Ribo' } { [-6.07] } { [-3] }

```
activeNames = {..., % param name, nominal value, range of parameters for uniform prior,
'TX_elong_glob' , exp(2.6), [exp(0) exp(5)] %3 from est params above
'TL_elong_glob' , exp(3.5), [exp(0) exp(6)] %2 from est params above
'AGTPdeg_time' , exp(8.8), [exp(6) exp(18)] %3 from est params above
'AGTPreg_ON' , exp(-3.9), [exp(-6) exp(-1)] %4 fixed in mcmc_info_vnprl_F2
'AGTPdeg_Rate' , exp(-9.9), [exp(-13) exp(-7)] %5 from est params above
'TXTL_UTR_UTR1_Kd' , exp(11), [exp(-3) exp(15)] %6 from est params above
'TXTL_PTET_RNAPbound_Kd' , exp(14), [exp(0) exp(17)] %7 TO BE ESTIMATED HERE
'TXTL_PTET_RNAPbound_F' , exp(1.5), [exp(0) exp(4)] %8 fixed in mcmc_info_vnprl_F2
'TXTL_INDUCER_TETR_ATC_Kd' , exp(2.9), [exp(0) exp(5)] %9 fixed in mcmc_info_vnprl_F2
'TXTL_DIMER_tetR_F' , exp(14), [exp(10) exp(20)] %10 fixed in mcmc_info_vnprl_F2
'TXTL_DIMER_tetR_Kd' , exp(12), [exp(3) exp(15)] %11 TO BE ESTIMATED HERE
'TXTL_PTET_sequestration_Kd' , exp(1.5), [exp(-2) exp(5)] %12 TO BE ESTIMATED HERE
'TL_AA_Kd' , exp(6.6), [exp(3) exp(10)] %13 fixed in mcmc_info_vnprl_F2
'TL_AGT_P_Kd' , exp(14.5), [exp(10) exp(18)] %14 fixed in mcmc_info_vnprl_F2
'TXTL_RNAdeg_Kd' , exp(15.2), [exp(7) exp(17)] %15 from est params above
'TXTL_INDUCER_TETR_ATC_Kd' , exp(13), [exp(0) exp(18)] %16 TO BE ESTIMATED HERE
'TXTL_DIMER_tetR_F' , exp(2.6), [exp(-2) exp(5)] %17 TO BE ESTIMATED HERE
'TXTL_DIMER_tetR_Kd' , exp(13), [exp(7) exp(17)] %18 TO BE ESTIMATED HERE
'TXTL_DIMER_tetR_F' , exp(2.6), [exp(-2) exp(5)] %19 TO BE ESTIMATED HERE
'TXTL_UTR_UTR1_F' , exp(-1.2), [exp(-4) exp(2)] %20 fixed in mcmc_info_vnprl_F2
'TXTL_PTET_RNAPbound_Kd' , exp(13.8), [exp(5) exp(17)] %21 from est params above
'TXTL_PLAC_RNAPbound_F' , exp(2.6), [exp(-2) exp(5)] %22 fixed in mcmc_info_vnprl_F2
'TXTL_RNAPBOUND_TERMINATION_RATE' , exp(1.8), [exp(-3) exp(6)] %23 from est params above
'TXTL_NTP_RNAP_1_F' , exp(0), [exp(-2) exp(3)] %24 fixed in mcmc_info_vnprl_F2
'TXTL_NTP_RNAP_2_F' , exp(0), [exp(-2) exp(3)] %25 fixed in mcmc_info_vnprl_F2
'TL_AA_F' , exp(-0.3), [exp(-3) exp(3)] %26 fixed in mcmc_info_vnprl_F2
'TL_AGT_F' , exp(-1.2), [exp(-4) exp(2)] %27 fixed in mcmc_info_vnprl_F2
'TXTL_RIBOBOUND_TERMINATION_RATE' , exp(2.3), [exp(0) exp(5)] %28 from est params above
'TXTL_RNAdeg_F' , exp(0), [exp(-3) exp(3)] %29 fixed in mcmc_info_vnprl_F2
'TXTL_RNAdeg_Kc' , exp(-0.45), [exp(-5) exp(3)] %30 from est params above
'RNAP' , exp(1.4419), [exp(1) exp(15)] %31 31% from est params above
'RNase' , exp(8.5), [exp(5) exp(12)] %32 from est params above
'Ribo' , exp(3.75), [exp(1) exp(12)] %33 from est params above
'TXTL_PROT_deGFP_MATURATION' , exp(-6.07), [exp(-9) exp(-3)] %34 fixed in mcmc_info_vnprl_F2
```

```
% Set the master vector values that are set from the values estimated in "vnprl_F2"
activeNames(indicesMasterVectorEstimated, 2) = num2cell(exp(paramVecToUse));
```

```
preFixedParams = {...
4 'AGTPreg_ON' , exp( -3.9120)
34 'TXTL_PROT_deGFP_MATURATION' , exp( -6.0748)
8 'TXTL_PTET_RNAPbound_F' , exp( 1.5000)
9 'TXTL_NTP_RNAP_1_Kd' , exp( 2.9459)
10 'TXTL_NTP_RNAP_2_Kd' , exp( 13.9970)
13 'TL_AA_Kd' , exp( 6.5566)
14 'TL_AGT_P_Kd' , exp( 14.5090)
20 'TXTL_UTR_UTR1_F' , exp( -0.2000)
22 'TXTL_PLAC_RNAPbound_F' , exp( 1.5000)
24 'TXTL_NTP_RNAP_1_F' , exp( 0)
25 'TXTL_NTP_RNAP_2_F' , exp( 0)
26 'TL_AA_F' , exp( -0.3000)
27 'TL_AGT_F' , exp( -1.2000)
29 'TXTL_RNAdeg_F' , exp( 0)}; % checked and verified.
```

```
mtet_phase1_params = ...
{...
7 , 'TXTL_PTET_RNAPbound_Kd' , exp(14) , [exp(0) exp(25)]
11 , 'TXTL_PTET_sequestration_Kd' , exp(-1) , [exp(-10) exp(5)]
12 , 'TXTL_PTET_sequestration_F' , exp(1.314) , [exp(-2) exp(5)]
16 , 'TXTL_INDUCER_TETR_ATC_Kd' , exp(-2) , [exp(-15) exp(5)]
17 , 'TXTL_INDUCER_TETR_ATC_F' , exp(1.577) , [exp(-2) exp(5)]
18 , 'TXTL_DIMER_tetR_Kd' , exp(-10) , [exp(-20) exp(-7)]
19 , 'TXTL_DIMER_tetR_F' , exp(1.447) , [exp(-2) exp(5)]...}
```

```

activeNames(cell2mat(mttet_phase1_params(:,1)),2) = mttet_phase1_params(:,3);
activeNames(cell2mat(mttet_phase1_params(:,1)),3) = mttet_phase1_params(:,4);
% Thus, the 10 estimated parameters are:

% 'TX_elong_glob' , exp(2.6), [exp(0) exp(5)] %1 % from est params above
% 'TL_elong_glob' , exp(3.5), [exp(0) exp(6)] %2 % from est params above
% 'AGTPdeg_time' , exp(8.8), [exp(6) exp(11)] %3 % from est params above
% 'TXTL_Pdeg_Kd' , exp(14), [exp(0) exp(17)] %7 % TO BE ESTIMATED HERE
% 'TXTL_PTE1_sequestration_Kd' , exp(12), [exp(3) exp(15)] %11 % TO BE ESTIMATED HERE
% 'TXTL_INDUCER_TETR_ATC_Kd' , exp(13), [exp(0) exp(18)] %16 % TO BE ESTIMATED HERE
% 'TXTL_PLAC_RNAPbound_Kd' , exp(13.8), [exp(5) exp(17)] %21 from est params above
% 'RNAP' , exp(1.4419), [exp(-1) exp(4)] %31 31% from est params above
% 'RNase' , exp(8.5), [exp(5) exp(10)] %32 from est params above
% 'Ribo' , exp(3.75), [exp(1) exp(6)] %33 % from est params above

```

```

estParamsIX = [1 2 3 7 11 16 21 31 32 33]';
estParams = activeNames(estParamsIX,1);
activeNames
% skipping AGTPdeo rate. AGTPdeo ON. TXTL PROT deGFP MATURATION

```

```

tsIDtouse = 4;
plotflag = true;
switch tsIDtouse

case 1
    ts1 = '20190427_170334_1_2058';
    ts2 = '20190427_170334_2_1029';

    tstamp = {ts1 ts2};
    niterID = {1:10 1:4};
    load(projdir '/simdata/' ts1 '/full_variable_set_' ts1 '.mat', ...
        'm1',...
        'mcmc_info', 'data_info', 'mai', 'ri');

case 2
    ts1 = '20190428_142033_1_2058';
    ts2 = '20190428_142033_2_1029';
    ts3 = '20190429_083138_1_1029';
    ts4 = '20190429_200219_1_1029';
    ts5 = '20190429_200219_2_412';
    ts6 = '20190430_141254_1_412';
    ts7 = '20190430_141254_2_206';
    ts8 = '20190501_042800_1_1029';
    ts9 = '20190501_105714_1_412';
    ts10 = '20190502_105714_1_412';
    ts11 = '20190503_073640_1_412';
    ts12 = '20190503_113414_1_412';
    ts13 = '20190503_170440_1_412';
    ts14 = '20190504_155250_1_412';
    tstamp = {ts1 ts2 ts3 ts4 ts5 ts6 ts7 ts8 ts9 ts10 ts11 ts12 ts13 ts14};
    niterID = {1:10 1:2 1:4 1:5 1:3 1:5 1:5 1:4 1:8 1:11 1 1 1 11 13};
    load(projdir '/simdata/' ts1 '/full_variable_set_' ts1 '.mat', ...
        'm1',...
        'mcmc_info', 'data_info', 'mai', 'ri');

end
tsToSave = ts14;
mai.masterVector

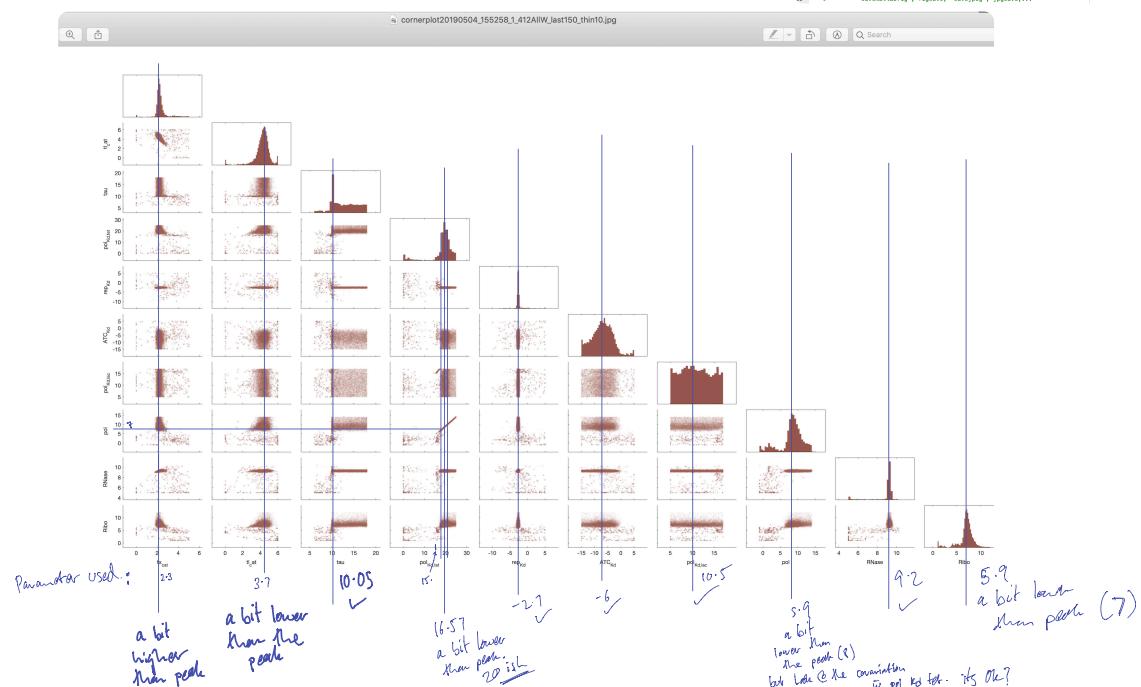
```

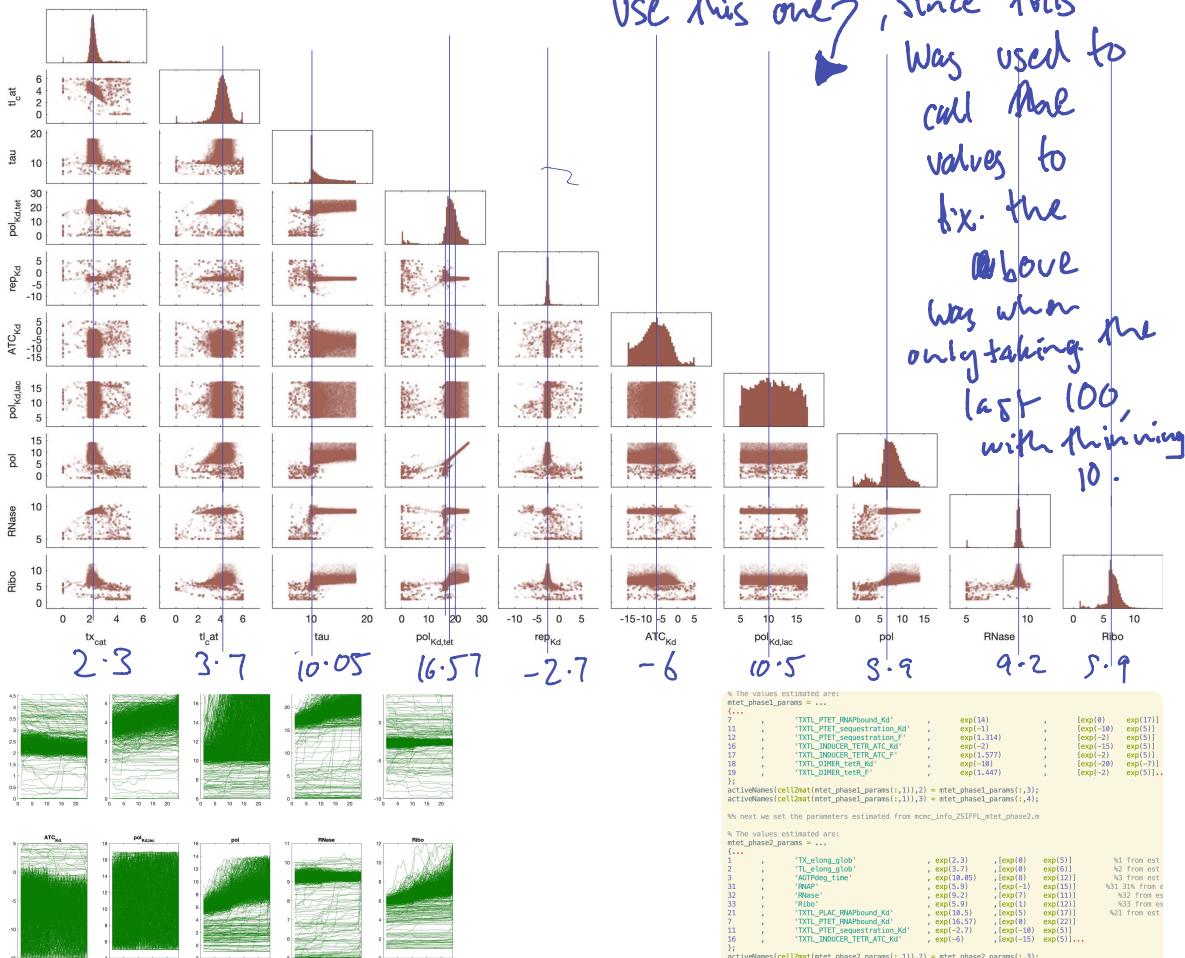
1aa864160f23c1ef7b89ea1412cee7d9006aeaab

```

91 % {'Kobs'}
92 % if plotflag
93 %    mode_plt(m, l, end, lenc-1800)(1800), paramset());
94 %    mode_plt(m, l, end, lenc), figsave, savefig, jpsave, ...
95 %    'projdir', projdir, 'tstamp', tsIDtouse, 'netalignparamtrg', 'AllW_lattice_main');
96 %    'wcc_plt(m, l, end, lenc), paramset(), ...
97 %    savefig, figsave, 'wcc_lattice_main');
98 % end

```





```
% The values estimated are:
met_phase1_params = ...
{...
  7 , 'TXTL_PET_PMAPbound_Kd' , , exp(14) , , [exp(8) exp(17)]
11 , 'TXTL_PET_sequestration_Kd' , , exp(-1) , , [exp(-18) exp(5)]
12 , 'TXTL_PET_sequestration_F' , , exp(1,214) , , [exp(-2) exp(5)]
16 , 'TXTL_INDUCER_TETR_ATC_F' , , exp(1,277) , , [exp(-2) exp(5)]
17 , 'TXTL_INDUCER_TETR_ATC_Kd' , , exp(1,577) , , [exp(-2) exp(5)]
18 , 'TXTL_DIMER_RecF' , , exp(1,441) , , [exp(-2) exp(7)]
19 , 'TXTL_DIMER_RecKd' , , exp(1,447) , , [exp(-2) exp(9)]};
activeNames(cell12m1(met_phase1_params(:,1),2)) = met_phase1_params(:,3);
activeNames(cell12m1(met_phase1_params(:,1),3)) = met_phase1_params(:,4);

% next we set the parameters estimated from mcmc_info_ZSIFFL_mtet_phase2.n
% The values estimated are:
met_phase2_params = ...
{...
  1 , 'TX_elong_glob' , , exp(2,0) , , [exp(8) exp(5)] , %1 from est
2 , 'TL_elong_glob' , , exp(3,7) , , [exp(8) exp(6)] , %2 from est
3 , 'ActProg_time' , , exp(18,89) , , [exp(8) exp(12)] , %3 from est
31 , 'Rnase' , , exp(9,2) , , [exp(7) exp(11)] , %32 from es
32 , 'Rnase' , , exp(9,2) , , [exp(7) exp(11)] , %33 from es
33 , 'Rnase' , , exp(9,2) , , [exp(7) exp(11)] , %34 from es
21 , 'TXTL_PLAC_PMAPbound_Kd' , , exp(18,5) , , [exp(8) exp(17)] , %35 from es
7 , 'TXTL_PET_PMAPbound_Kd' , , exp(16,577) , , [exp(8) exp(22)] , %36 from es
13 , 'TXTL_INDUCER_TETR_ATC_F' , , exp(1,277) , , [exp(-2) exp(5)] , %37 from es
16 , 'TXTL_INDUCER_TETR_ATC_Kd' , , exp(-8) , , [exp(-15) exp(5)]...};

activeNames(cell12m1(met_phase2_params(:,1),2)) = met_phase2_params(:,3);
activeNames(cell12m1(met_phase2_params(:,1),3)) = met_phase2_params(:,4);

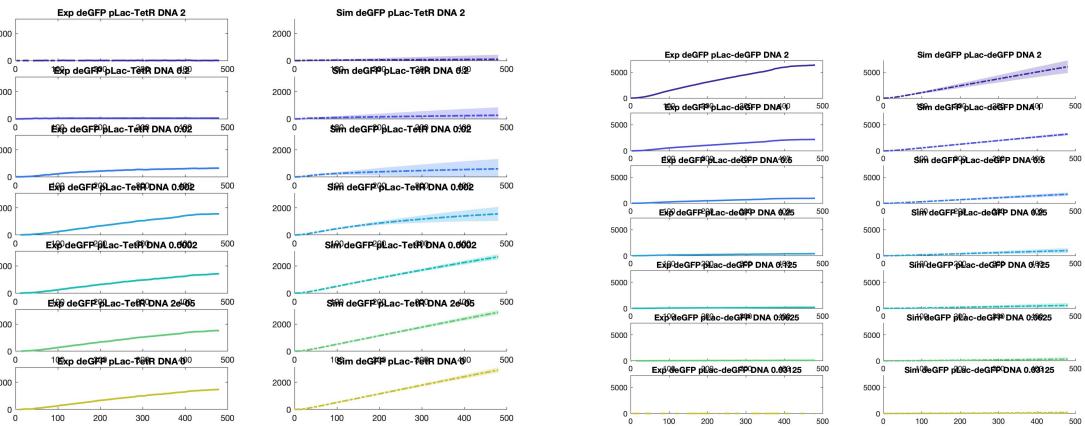
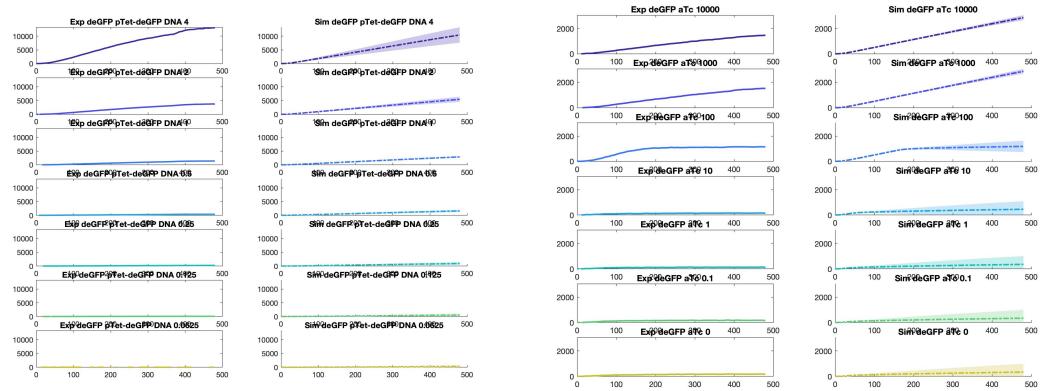
% we also set the forward rate parameters, since those should not matter much, and any value around 1 is
% the F rate parameters just set the timescale.

met_phase2_params = ...
{...
  36 , 'TXTL_INDUCER_LASR_AHL_F' , , exp(0) , , [exp(-3) exp(5)]
38 , 'TXTL_PLAS_RNAPbound_F' , , exp(0) , , [exp(-3) exp(5)]
41 , 'TXTL_PLAS_RNAPbound_F' , , exp(0) , , [exp(-3) exp(5)]
42 , 'TXTL_PLAS_TRANS_F' , , exp(0) , , [exp(-3) exp(5)]};

activeNames(cell12m1(met_phase2_params(:,1),2)) = met_phase2_params(:,3);
activeNames(cell12m1(met_phase2_params(:,1),3)) = met_phase2_params(:,4);
```

/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_mtet_phase2/simdata_20190504_155258_1_412

This went through a few iterations of parameter range expansion. commits
[da77c5446bd97c422207b3d5854fd70206ce6c8](#) to [27cb38b91e8b96547a4ee9e91f5188a26bba0800](#)



```

case 1
ts1 = '28190001_070000_1_180';
ts2 = '28190001_123454_1_180';
ts3 = '28190001_123454_2_180';
ts4 = '28190001_123454_3_180';
ts5 = '28190001_128418_1_180';

tstamp = [ts1 ts2 ts3 ts4 ts5];
nterID = {1 15 15 1 12 12};
loadFile = 'simdata_1.tsl' 'full_variable_set_'.ts1 '.mat'; ...
'nter_id',...
'mcmc_info', 'data_info', 'mai', 'rl1';

% The values estimated are:
mtet_phasel_params = ...
{...
7 , 'TXTL_PTET_RNAPbound_Kd' , exp(14) , [exp(0) exp(5)] %1 from est
11 , 'TXTL_PTET_sequestration_Kd' , exp(-1) , [exp(-10) exp(5)] %2 from est
12 , 'TXTL_PTET_sequestration_F' , exp(1.314) , [exp(-2) exp(5)] %3 from est
16 , 'TXTL_DIMER_tetR_AHL_Kd' , exp(0.001) , [exp(0) exp(5)] %31 31% from est
17 , 'TXTL_INDUCER_TETR_ATC_F' , exp(1.577) , [exp(-2) exp(5)] %33 from est
18 , 'TXTL_DIMER_tetKd' , exp(-10) , [exp(-20) exp(-7)] %33 from est
19 , 'TXTL_DIMER_tetF' , exp(1.447) , [exp(-2) exp(5)]..

activeNames(cell2mat(mtet_phase1_params(:,1)),2) = mtet_phase1_params(:,3);
activeNames(cell2mat(mtet_phase1_params(:,1)),3) = mtet_phase1_params(:,4);

% next we set the parameters estimated from mcmc_info_ZSIFFL_mtet_phase2.m
% The values estimated:
mtet_phase2_params = ...
{...
1 , 'RIBO_elong_glob' , exp(2.3) , [exp(0) exp(5)] %1 from est
2 , 'RIBO_elong_glob' , exp(3.7) , [exp(0) exp(6)] %2 from est
3 , 'AGTPdeq_time' , exp(10.05) , [exp(0) exp(12)] %3 from est
31 , 'RNAP' , exp(5.9) , [exp(-1) exp(15)] %31 31% from est
32 , 'RIBO' , exp(0.001) , [exp(0) exp(5)] %33 from est
33 , 'RIBO' , exp(5.9) , [exp(1) exp(12)] %33 from est
21 , 'TXTL_PTET_RNAPbound_Kd' , exp(10.57) , [exp(0) exp(17)] %21 from est
7 , 'TXTL_PTET_RNAPbound_F' , exp(16.57) , [exp(0) exp(22)] %21 from est
11 , 'TXTL_PTET_sequestration_Kd' , exp(-2.7) , [exp(-10) exp(5)] %21 from est
16 , 'TXTL_INDUCER_TETR_ATC_Kd' , exp(-6) , [exp(-15) exp(5)]..

activeNames(cell2mat(mtet_phase2_params(:,1)),2) = mtet_phase2_params(:,3);
activeNames(cell2mat(mtet_phase2_params(:,1)),3) = mtet_phase2_params(:,4);

% we also set the forward rate parameters, since those should not matter much, and any value around 1 is
% the f_rate parameter, so just set the timescale.
mtet_phase2_params = ...
{...
36 , 'TXTL_INDUCER_LASR_AHL_F' , exp(0) , [exp(-3) exp(5)] %1 from est
38 , 'TXTL_PLAS_RNAPbound_F' , exp(0) , [exp(0) exp(5)] %2 from est
41 , 'TXTL_PLAS_TFBIND_R' , exp(0) , [exp(0) exp(5)] %3 from est
42 , 'TXTL_PLAS_TFBIND_F' , exp(0) , [exp(-3) exp(5)]..

activeNames(cell2mat(mtet_phase2_params(:,1)),2) = mtet_phase2_params(:,3);
activeNames(cell2mat(mtet_phase2_params(:,1)),3) = mtet_phase2_params(:,4);

% Set the master vector values that are set from the values estimated in 'vn
activeNames(indicesMasterVectorEstimated, 2) = num2cell(exp(paramVecToUse));

```

```

% Set the master vector values that were already fixed in 'vn
prefixedParams = {...;
4 'AGTPdeq_time' , exp(-3.9120)
34 'TXTL_PTET_deGFP_MATURATION' , exp(-6.0748)
8 'TXTL_PTET_RNAPbound_F' , exp(1.5000)
9 'TXTL_NTP_RNAP_1_Kd' , exp(2.9459)
10 'TXTL_NTP_RNAP_2_Kd' , exp(13.9970)
13 'TL_AA_Kd' , exp(6.5566)
14 'TL_AGP_Kd' , exp(14.5090)
20 'TXTL_UTR_UTR1_F' , exp(-0.2000)
22 'TXTL_PLAC_RNAPbound_F' , exp(1.5000)
24 'TXTL_NTP_RNAP_1_F' , exp( 0)
25 'TXTL_NTP_RNAP_2_F' , exp( 0)
26 'TL_AA_F' , exp(-0.3000)
27 'TL_AGP_F' , exp(-1.2000)
29 'TXTL_NNAdeg_F' , exp( 0)); % checked and ve

```

```

paramnames = ...
{...
{'pol_{Kd,tet}'}
{'pol_{Kd,lac}'}
{'pol'}
{'Ribo'}
{'3OC12_{Kd}'}
{'3OC12_{F}'}
{'pol_{Kd,las}'}
{'pol_{F,las}'}
{'plas_{tf, Kd}'}
{'plas-pol_{tf, Kd}'}
{'plas-pol_{tf, F}'}
{'plas_{tf, F}'}
};

% activeNames(estParams)

```

Reaction: [protein lasR] + OC12HSL <-> [OC12HSL:protein lasR]
 ReactionRate: [TXTL_INDUCER_LASR_AHL_F](#)[protein lasR]*OC12HSL - TXTL_INDUCER_LASR_AHL_R*[OC12HSL:protein lasR]

Reaction: [DNA plas--utr1--deGFP] + RNAP <-> [RNAP:DNA plas--utr1--deGFP]
 ReactionRate: [TXTL_PLAS_RNAPbound_F](#)*[DNA plas--utr1--deGFP]*RNAP - TXTL_PLAS_RNAPbound_R*[RNAP:DNA plas--utr1--deGFP]

Reaction: [DNA plas--utr1--deGFP] + [OC12HSL:protein lasR] <-> [DNA plas--utr1--deGFP:OC12HSL:protein lasR]
 ReactionRate: [TXTL_PLAS_TFBIND_F](#)*[DNA plas--utr1--deGFP]*[OC12HSL:protein lasR] - TXTL_PLAS_TFBIND_R*[DNA plas--utr1--deGFP:OC12HSL:protein lasR]

Reaction: [DNA plas--utr1--deGFP:OC12HSL:protein lasR] + RNAP <-> [RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]
 ReactionRate: [TXTL_PLAS_TFRNAPbound_F](#)*[DNA plas--utr1--deGFP:OC12HSL:protein lasR]*RNAP - TXTL_PLAS_TFRNAPbound_R*[RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]

Reaction: [RNAP:DNA plas--utr1--deGFP] + [OC12HSL:protein lasR] <-> [RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]
 ReactionRate: [TXTL_PLAS_TFBIND_F](#)*[RNAP:DNA plas--utr1--deGFP]*[OC12HSL:protein lasR] - TXTL_PLAS_TFBIND_R*[RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]

• why the hell are these named the same way??? i think it does not matter, since i update activeNames with them, but still, this is just an error waiting to happen.

Dont know this was ever estimated. We fixed from the moment they were introduced

WAS THIS THE
ORIGINAL
TRAINING
FULL ???



Overall summary of what was estimated where and to what values.

- training full, before May 6 commit.

```
ans =
1x3 cell array
(1) ('TXTL_PTEL_RNAPbound_Kd') {[1.5713e+07]} %> double
('TXTL_PLAC_RNAPbound_Kd') {[3.6316e+04]} %> double
('RNAP') {[365.0375]} %> double
('Ribo') {[365.0375]} %> double
('TXTL_INDUCER_LASR_AHL_Kd') {[0.1353]} %> double
('TXTL_INDUCER_LASR_AHL_F') {[3.6693]} %> double
('TXTL_PLAS_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAS_RNAPbound_F') {[3.6693]} %> double
('TXTL_PLAS_TFBIND_Kd') {[0.1353]} %> double
('TXTL_PLAS_TFRNAPbound_Kd') {[7.3891]} %> double
('TXTL_PLAS_TFRNAPbound_F') {[3.6693]} %> double
('TXTL_PLAS_TFBIND_F') {[3.6693]} %> double
```

f2 >> |

training full

(2)

training full B

(3)

training full C

estParamsIX = [23 28 31 33 35 37 39 48];
estParams = activeNames(estParamsIX,1)

estParams =

```
1x3 cell array
(1) ('TXTL_PTEL_RNAPbound_Kd') {[1.5713e+07]} %> double
('TXTL_PLAC_RNAPbound_Kd') {[3.6316e+04]} %> double
('RNAP') {[365.0375]} %> double
('Ribo') {[365.0375]} %> double
('TXTL_INDUCER_LASR_AHL_Kd') {[0.1353]} %> double
('TXTL_INDUCER_LASR_AHL_F') {[3.6693]} %> double
('TXTL_PLAS_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAS_RNAPbound_F') {[3.6693]} %> double
('TXTL_PLAS_TFBIND_Kd') {[0.1353]} %> double
('TXTL_PLAS_TFRNAPbound_Kd') {[7.3891]} %> double
('TXTL_PLAS_TFRNAPbound_F') {[3.6693]} %> double
('TXTL_PLAS_TFBIND_F') {[3.6693]} %> double
```

Fitting did not work

Don't need to put into paper.

estParamsIX = [1 2 7 11 16 21 23 28 31 33 35 37 39 48];
estParams = activeNames(estParamsIX,1)

estParams =

```
1x3 cell array
(1) ('TX_elong_glob') {[1.5713e+07]} %> double
('TL_elong_glob') {[3.6316e+04]} %> double
('TXTL_PTEL_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PTEL_sequestration_Kd') {[0.1353]} %> double
('TXTL_RIBOBOUND_TERMINATION_RATE') {[0.1353]} %> double
('TXTL_PLAC_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAC_RNAPbound_TERMINATION_RATE') {[0.1353]} %> double
('TXTL_RIBOBOUND_TERMINATION_RATE') {[0.1353]} %> double
('RNAP') {[365.0375]} %> double
('Ribo') {[365.0375]} %> double
('TXTL_INDUCER_LASR_AHL_Kd') {[0.1353]} %> double
('TXTL_INDUCER_LASR_AHL_F') {[3.6693]} %> double
('TXTL_PLAS_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAS_RNAPbound_F') {[3.6693]} %> double
('TXTL_PLAS_TFBIND_Kd') {[0.1353]} %> double
('TXTL_PLAS_TFRNAPbound_Kd') {[7.3891]} %> double
('TXTL_PLAS_TFRNAPbound_F') {[3.6693]} %> double
```

>> |

Fixed in D?
or just in E?

training full E.

training full D

in addition those fixed in C.

estParamsIX = [1 2 7 21 23 28 31 33 35 37 39 48];
estParams = activeNames(estParamsIX,1)

estParams =

```
1x3 cell array
(1) ('TX_elong_glob') {[1.5713e+07]} %> double
('TL_elong_glob') {[3.6316e+04]} %> double
('TXTL_PTEL_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAC_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_RIBOBOUND_TERMINATION_RATE') {[0.1353]} %> double
('RNAP') {[365.0375]} %> double
('Ribo') {[365.0375]} %> double
('TXTL_INDUCER_LASR_AHL_Kd') {[0.1353]} %> double
('TXTL_INDUCER_LASR_AHL_F') {[3.6693]} %> double
('TXTL_PLAS_RNAPbound_Kd') {[0.1353]} %> double
('TXTL_PLAS_TFBIND_Kd') {[0.1353]} %> double
('TXTL_PLAS_TFRNAPbound_Kd') {[7.3891]} %> double
('TXTL_PLAS_TFRNAPbound_F') {[3.6693]} %> double
```

>> |

(5)

In the paper SI:

- Ph 1(VNPRL)&
- Ph 2amtel1&
- Ph 2bmtel2&
- Ph 2c (1) &
- Ph 2d (4) &
- Ph 2e (5) &
- & Ph 2f (6)

```
mte_phase2_params = ...
(...
  'TXTL_INDUCER_LASR_AHL_F' , exp(0) , [exp(-3) exp(5)]
38   , 'TXTL_PLAS_RNAPbound_F' , exp(0) , [exp(-3) exp(5)]
40   , 'TXTL_RIBOBOUND_TERMINATION_RATE' , exp(0) , [exp(-2) exp(1)]
41   , 'TXTL_INDUCER_TETR_ATC_Kd' , exp(13) , [exp(5) exp(15)] ...
42   , 'TXTL_PLAS_TFBIND_F' , exp(0) , [exp(-3) exp(5)] );
activeNames(cell2mat(mte_phase2_params(:,1)),2) = mte_phase2_params(:,3);
activeNames(cell2mat(mte_phase2_params(:,1)),3) = mte_phase2_params(:,4);

training_full_params = ...
(...
  'TX_elong_glob' , exp(3.121) , [exp(1) exp(6)] %1 from est params above
1   , 'TL_elong_glob' , exp(3.436) , [exp(1) exp(6)] %2 from est params above
2 );
activeNames(cell2mat(training_full_params(:,1)),2) = training_full_params(:,3);
activeNames(cell2mat(training_full_params(:,1)),3) = training_full_params(:,4);

estParamsIX = [7 21 23 28 31 33 37 39 48];
estParams = activeNames(estParamsIX,1)
estParams =
```

```
1x3 cell array
('TXTL_PTEL_RNAPbound_Kd')
('TXTL_PLAC_RNAPbound_Kd')
('TXTL_RIBOBOUND_TERMINATION_RATE')
('TXTL_RIBOBOUND_TERMINATION_RATE')
('Ribo')
('TXTL_INDUCER_LASR_AHL_Kd')
('TXTL_PLAS_RNAPbound_Kd')
('TXTL_PLAS_TFBIND_Kd')
('TXTL_PLAS_TFRNAPbound_Kd')
```

(6)

Can just say that we started with (0), found that the protein expression levels were not high enough.
introduced termination rates,
& totR params, but fixed the
F rates. (to reduce the # of
params) to get to (4).
Then we fixed 3 more
to get to (5). Then we fixed
2 more to get to (6).

Show the plots for
(5) (D) & (6) (E).

Build a table with
params.

	(0)	(4)	(5)	(6)
totR params				

There are two DIFFERENT training full's:
 training full -- pre May 6 commit
 training full post may 6 commit

I guess after this i realized that i cannot just change the training full file, and started making A, B, C and so on.

Anyway, this is what the data looks like:

/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsm_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_training/simdata_20190505_023615_1_738/

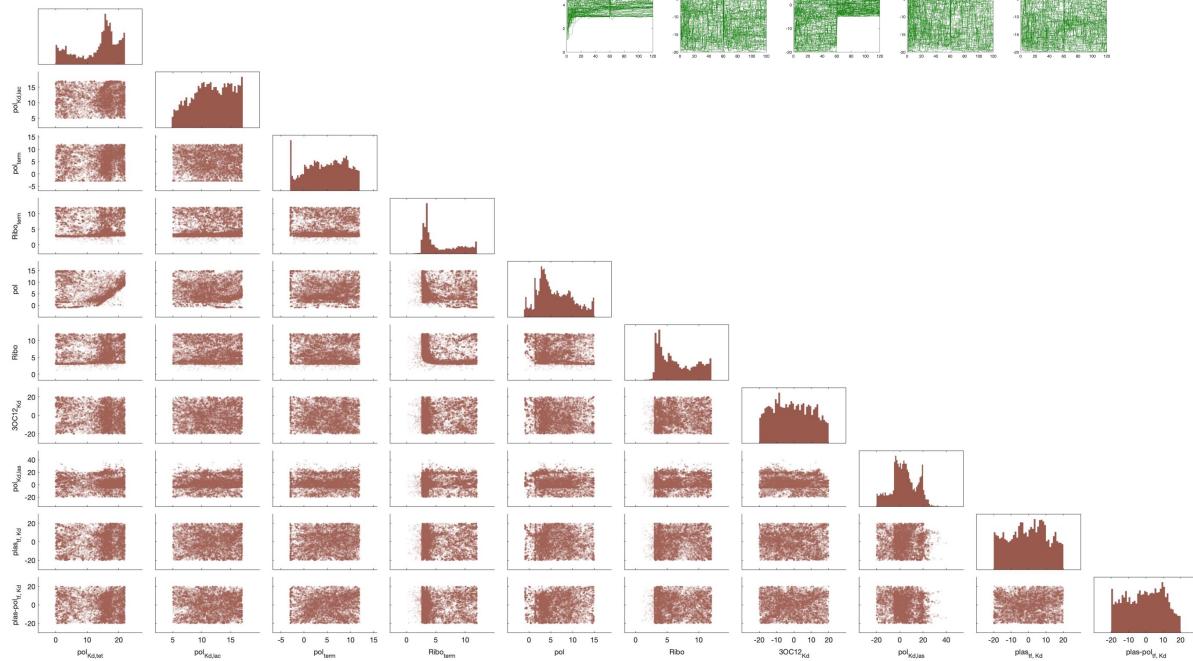
```
ans =
12x3 cell array
{ 'TXTL_PTEL_RNAPbound_Kd' } { [1.5713e+07] } { 1x2 double }
{ 'TXTL_PPLAC_RNAPbound_Kd' } { [3.6305e+04] } { 1x2 double }
{ 'RMAP' } { [ 365.0375] } { 1x2 double }
{ 'Ribo' } { [ 365.0375] } { 1x2 double }
{ 'TXTL_INDUCER_LASR AHL_Kd' } { [ 0.1353] } { 1x2 double }
* { 'TXTL_INDUCER_LASR AHL_F' } { [ 3.6693] } { 1x2 double }
* { 'TXTL_PLAS_RNAPbound_Kd' } { [ 0.1353] } { 1x2 double }
* { 'TXTL_PLAS_TFRNAPbound_F' } { [ 3.6693] } { 1x2 double }
* { 'TXTL_PLAS_TFRNAPbound_Kd' } { [ 7.3891] } { 1x2 double }
* { 'TXTL_PLAS_TFRNAPbound_F' } { [ 3.6693] } { 1x2 double }
* { 'TXTL_PLAS_TFBIND_F' } { [ 3.6693] } { 1x2 double }
```



ZSIFFL_training_full -- AFTER the may 6 commit

/Users/vipulsinghal/Dropbox/
Documents/toolbox/
txtlsm_vsfork2017/mcmc_simbio/
projects/proj_ZSIFFL_training/
simdata_20190507_102955_1_1845/
cornerplot20190507_102955_1_1845
Burned_in.jpg

/Users/vipulsinghal/Dropbox/
Documents/toolbox/
txtlsm_vsfork2017/mcmc_simbio/
projects/proj_ZSIFFL_training/
simdata_20190507_102955_1_1845/
trace20190507_102955_1_1845Burn
ed_in.jpg



```
estParamsIX = [7 21 23 26 31 33 35 37 39 48];
estParams = activebases(estParamsIX,1);
activebases(estParamsPA,1)
```

```
ans =
```

```
1x3 cell array
```

```
{'TXTL_PET_RNAbound_Kd' } {{1.5713e+07}} {1x2 double}
{'TXTL_PLAC_RNAbound_Kd' } {{3.6316e+04}} {1x2 double}
{'TXTL_RNABOUND_TERMINATION_RATE' } {{ 1.76e-05}} {1x2 double}
{'TXTL_RIBOBOUND_TERMINATION_RATE' } {{ 0.0009391}} {1x2 double}
{'RNAP' } {{ 0.0003751}} {1x2 double}
{'TXTL_INDUCER_LASR_AHL_Kd' } {{ 365.8075}} {1x2 double}
{'TXTL_PLAC_RNAbound_Kd' } {{1.8686e+13}} {1x2 double}
{'TXTL_PLAS_TFIND_Kd' } {{ 0.13531}} {1x2 double}
{'TXTL_PLAS_TFNAPbound_Kd' } {{ 7.3891}} {1x2 double}
```

allowing the
termination
rates to very
very
this still
needs to go
high now!

in full B
below,
I fix the
plot & place
KD &
just try to
estimate 8
params.

(lower =
higher
chance of
fitting)

(0-2=8
in the next
full B
(next)
Sims.)

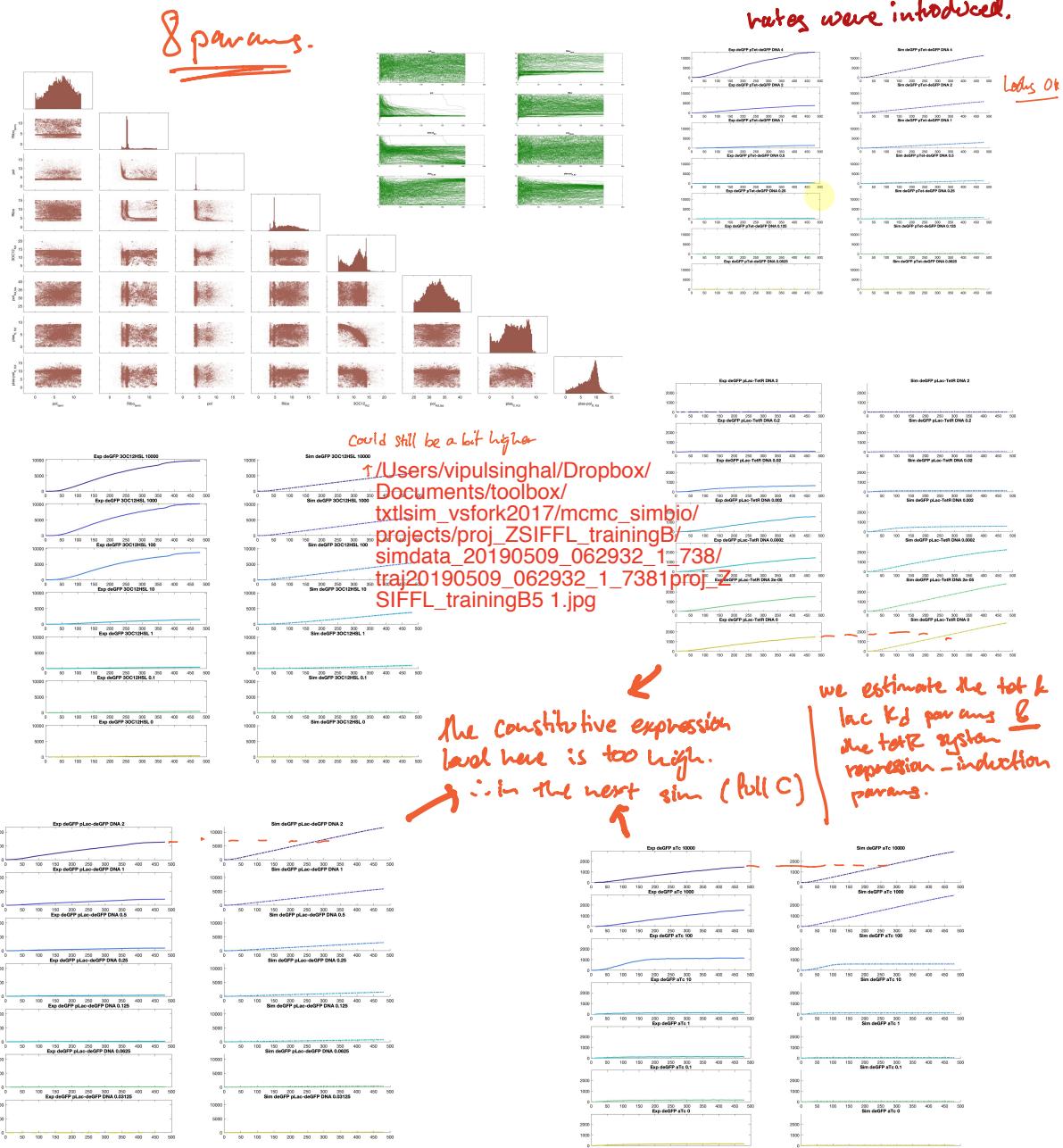
ZSIFFL_training_fullB

here I thought that maybe the fact that the termination rates for the rnap and ribo were fixed was causing a bottleneck in the expression levels, limiting the simulated gfp from the plas-gfp to be much lower than the experimental data.

Yup. I don't think the current list of params is the one I used for training-B. it was

long story short, In this sim, we estimate a larger set of params to try to get closer. And then in training C, that is where we really start to get close.

*only in training-B
that the termination
rates were introduced.*



full B

```
activeNames(cell2mat(mtet_phase1_params(:,1)),2) = mtet_phase1_params(:,3);
activeNames(cell2mat(mtet_phase1_params(:,1)),3) = mtet_phase1_params(:,4);

% next we set the parameters estimated from mcmc_info_2SIFFL_mtet_phase2.m

% The values estimated are:
mtet_phase2_params = ...
{...
1 , , 'TXL_elong_glob' , exp(2.3) , [exp(0) exp(5)]
2 , , 'TL_elong_glob' , exp(3.7) , [exp(0) exp(6)]
3 , , 'AGTDeg_time' , exp(10.05) , [exp(8) exp(12)]
51 , , 'RNAP' , exp(5.9) , [exp(1) exp(15)] %
52 , , 'Pulse' , exp(9.2) , [exp(7) exp(11)]
73 , , 'Ribo' , exp(5.0) , [exp(1) exp(15)]
21 , , 'TXTL_PLAC_RNAPbound_Kd' , exp(10.5) , [exp(5) exp(17)]
37 , , 'TXTL_PTET_RNAPbound_Kd' , exp(17) , [exp(0) exp(22)]
11 , , 'TXTL_PTET_sequestration_Kd' , exp(-2.7) , [exp(-10) exp(5)]
16 , , 'TXTL_INDUCER_TETR_ATC_Kd' , exp(-6) , [exp(-15) exp(5)]...
};

activeNames(cell2mat(mtet_phase2_params(:,1)),2) = mtet_phase2_params(:,3);
activeNames(cell2mat(mtet_phase2_params(:,1)),3) = mtet_phase2_params(:,4);

% we also set the forward rate parameters, since those should not matter much, and any val
% the F rate parameters just set the timescale.

mtet_phase2_params = ...
{...
36 , , 'TXTL_INDUCER_LASR_AHL_F' , exp(0) , [exp(-3) exp(5)]
38 , , 'TXTL_PLAS_RNAPbound_F' , exp(0) , [exp(-3) exp(5)]
41 , , 'TXTL_PLAS_TFRNAPbound_F' , exp(0) , [exp(-3) exp(5)]
42 , , 'TXTL_PLAS_TFBIND_F' , exp(0) , [exp(-3) exp(5)];
activeNames(cell2mat(mtet_phase2_params(:,1)),2) = mtet_phase2_params(:,3);
activeNames(cell2mat(mtet_phase2_params(:,1)),3) = mtet_phase2_params(:,4);

estParamsIX = [23 28 31 33 35 37 39 40];
estParams = activeNames(estParamsIX,1);
```

full B

```
tsIDtouse = 1;
plotflag = true;
switch tsIDtouse
    case 1 % this is after including the terminat.
        ts1 = '20190508_001705_1_1845';
        ts2 = '20190508_213950_1_1107';
        ts3 = '20190509_024244_1_738';
        ts4 = '20190509_062932_1_738';
        tstamp = {ts1 ts2 ts3 ts4};
        nIterID = {1:7 1:9 1:6 1:2};
        load([projdir '/simdata' ts1 '/full_vari
            'mi',...
            'mcmc_info', 'data_info', 'ri');
    end
    tsToSave = ts4;
    mai.masterVector
    marray_full = mcmc_get_walkers(tstamp,nIterID, pr
    marray = marray_full(:,:,1:end);
    clear marray_full
    parnames = ...
        {'pol_{term}'}
        {'Ribo_{term}'}
        {'pol'}
        {'Ribo'}
        {'3OC12_{Kd}'}
        {'pol_{Kd,las}'}
        {'plas_{tf, Kd}'}
        {'plas-pol_{tf, Kd}'};
    % activeNames(estParamsIX,:)
    %
    % ans =
%
```

EstimatedParams = [..]

2.5234	2.4464	2.6231	2.5976	2.4650	2.4277	2.4806	2.4601
8.8054	8.9024	8.8997	8.8014	8.8483	8.8642	8.8394	8.81
-10.0179	-9.7873	-9.9376	-10.0048	-9.9056	-9.8889	-9.9100	-9.1
-7.7320	13.6474	13.9213	13.9325	13.9277	13.9265	13.9272	13.1
2.4080	4.1790	0.8620	2.2727	1.5489	4.4090	1.1170	3.7
15.6977	15.6349	15.7098	15.6858	15.6661	15.6339	15.7103	15.4
0.8935	-0.2251	-0.1516	-0.0709	-0.2189	-0.1276	-0.0340	-0.1
1.5652	1.6167	1.4897	1.5206	1.6088	1.6259	1.6008	1.1
8.3976	8.6141	8.6149	8.5123	8.5710	8.5146	8.4985	8.1
3.6442	3.2731	3.2538	3.3828	3.2633	3.2936	3.2387	3.4
8.0100	0.0442	4.5196	-2.2731	-1.6399	-1.7370	-2.9377	0.1
1.5533	2.8328	2.8292	2.7929	2.9282	2.8804	2.9659	2.1
3.9583	4.1863	4.1261	4.0812	4.1922	3.9954	4.0318	3.1

ParamColumnToUse = 2;

paramVecToUse = EstimatedParams(:, ParamColumnToUse);

indicesMasterVectorEstimated = [1 3 5 21 23 15 30 31 32 2 6 28 33];

activeNames = { .., param name, nominal value, range of parameters for uniform }

'TX_elong_glob' , exp(2.6), [exp(0) exp(5)]
 'TL_elong_glob' , exp(3.5), [exp(0) exp(6)]
 'AGTPdeg_time' , exp(8.8), [exp(6) exp(11)]
 'AGTPreg_ON' , exp(-3.9), [exp(-6) exp(-1)]
 'AGTPdeg_rate1' , exp(-9.9), [exp(-13) exp(-7)]
 'AGTPdeg_rate2' , exp(-1.1), [exp(0) exp(15)]
 'TXL_UTR1_Kd' , exp(1.4), [exp(0) exp(17)]
 'TXL_PTET_RNAbound_Kd' , exp(1.5), [exp(0) exp(4)]
 'TXL_PTET_RNAbound_F' , exp(2.9), [exp(0) exp(5)]
 'TXL_NTP_RNAP_1_Kd' , exp(14), [exp(10) exp(20)]
 'TXL_NTP_RNAP_2_Kd' , exp(12), [exp(3) exp(15)]
 'TXL_PTET_sequestration_Kd' , exp(1.5), [exp(-2) exp(5)]
 'TXL_PTET_sequestration_F' , exp(1.8), [exp(0) exp(10)]
 'TL_AGP_Kd' , exp(1.5), [exp(10) exp(18)]
 'TXL_RNAdeg_Kd' , exp(15.2), [exp(7) exp(17)]
 'TXL_INDUCER_TETR_ATC_Kd' , exp(13), [exp(0) exp(18)]
 'TXL_INDUCER_TETR_ATC_F' , exp(2.6), [exp(-2) exp(5)]
 'TXL_DIMER_tet_Kd' , exp(13), [exp(7) exp(17)]
 'TXL_UTR1_F' , exp(2.6), [exp(-2) exp(5)]
 'TXL_UTR1_F' , exp(13.8), [exp(5) exp(17)]
 'TXL_PLAC_RNAbound_Kd' , exp(2.6), [exp(-2) exp(5)]
 'TXL_PLAC_RNAbound_F' , exp(1.8), [exp(0) exp(12)]
 'TXL_RNAbound_TERMINATION_RATE' , exp(0), [exp(-2) exp(3)]
 'TXL_NTP_RNAP_1_F' , exp(0), [exp(-2) exp(3)]
 'TXL_NTP_RNAP_2_F' , exp(0), [exp(-2) exp(3)]
 'TL_AGP_F' , exp(-1.2), [exp(-4) exp(3)]
 'TXL_RIBOROUND_TERMINATION_RATE' , exp(2.3), [exp(0) exp(12)]
 'TXL_RNAdeg_F' , exp(0), [exp(-3) exp(3)]
 'TXL_RNAdeg_Kc' , exp(-0.45), [exp(-5) exp(3)]
 'RNAP' , exp(1.4419), [exp(-1) exp(8)]
 'RNase' , exp(8.5), [exp(5) exp(18)]
 'RNApol' , exp(3.35), [exp(6) exp(6)]
 'TXL PROT_deGFP_MATURATION' , exp(-6.07), [exp(-9) exp(-3)]
 'TXL_INDUCER_LASR_AHL_Kd' , exp(-2), [exp(5) exp(20)] %36-1
 'TXL_INDUCER_LASR_AHL_F' , exp(0), [exp(-6) exp(6)] %37-1
 'TXL_PLAS_RNAbound_Kd' , exp(30), [exp(25) exp(40)] %38-1
 'TXL_PLAS_RNAbound_F' , exp(3), [exp(-6) exp(6)] %39-1 pol
 'TXL_PLAS_TFBIND_Kd' , exp(3), [exp(0) exp(10)] %40-1 pol
 'TXL_PLAS_TRNAbound_Kd' , exp(0), [exp(-6) exp(6)] %41-1 pol
 'TXL_PLAS_TRNAbound_F' , exp(0), [exp(-6) exp(6)] %42-1 pol
 'TXL_PLAS_TFBIND_F' , exp(0), [exp(-6) exp(6)] %43-1 pol

% Set the master vector values that are set from the values estimated in "vnpl_F2"

Set the master vector values that were already fixed in "vnpl_F2"

prefixedParams = { .. }

4	'AGTPreg_ON'	, exp(-3.9120)
34	'TXL_PROT_deGFP_MATURATION'	, exp(-6.0748)
8	'TXL_PTET_RNAbound_F'	, exp(1.5000)
9	'TXL_NTP_RNAP_1_Kd'	, exp(2.9459)
10	'TXL_NTP_RNAP_2_Kd'	, exp(13.9970)
13	'TL_AA_Kd'	, exp(6.5566)
14	'TL_AGP_Kd'	, exp(14.5090)
20	'TXL_UTR_UTR1_F'	, exp(-0.2000)
22	'TXL_PLAC_RNAbound_F'	, exp(1.5000)
24	'TXL_NTP_RNAP_1_F'	, exp(0)
25	'TXL_NTP_RNAP_2_F'	, exp(0)
26	'TL_AA_F'	, exp(-0.3000)
27	'TL_AGP_F'	, exp(-1.2000)
29	'TXL_RNAdeg_F'	, exp(0)); % checked a

% set the prefixed params elements in master vector to the values in

mter_phase1_params = { .. }

7	'TXL_PTET_RNAbound_Kd'	, exp(14)	, [exp(0) exp(17)]
11	'TXL_PTET_sequestration_Kd'	, exp(-1)	, [exp(-10) exp(5)]
12	'TXL_PTET_sequestration_F'	, exp(1.214)	, [exp(-2) exp(5)]
16	'TXL_INDUCER_TETR_ATC_Kd'	, exp(-2)	, [exp(-15) exp(5)]
17	'TXL_INDUCER_TETR_ATC_F'	, exp(1.577)	, [exp(-2) exp(5)]
18	'TXL_DIMER_tet_Kd'	, exp(-10)	, [exp(-20) exp(-7)]
19	'TXL_DIMER_tet_F'	, exp(14.447)	, [exp(-2) exp(5)];

activeNames(cell2mat(mter_phase1_params(:,1)),2) = mter_phase1_params(:,2);
 activeNames(cell2mat(mter_phase1_params(:,1)),3) = mter_phase1_params(:,3);

% next we set the parameters estimated from mcmc_info_ZSIFFL_mter_phase2.m

The values estimated are:

mter_phase2_params = { .. }

1	'TX_elong_glob'	, exp(2.3)	, [exp(0) exp(7)]	%1 from est
2	'TL_elong_glob'	, exp(3.7)	, [exp(0) exp(7)]	%2 from est
3	'AGTPdeg_time'	, exp(18.05)	, [exp(8) exp(12)]	%3 from est
31	'RNAP'	, exp(5.9)	, [exp(1) exp(15)]	%31 31% from est
32	'RNase'	, exp(1.2)	, [exp(1) exp(15)]	%32 from est
33	'Riboz'	, exp(5.9)	, [exp(1) exp(15)]	%33 from est
21	'TXL_PLAC_RNAbound_Kd'	, exp(10.5)	, [exp(5) exp(25)]	%21 from est
7	'TXL_PTET_RNAbound_Kd'	, exp(17)	, [exp(5) exp(25)]	
11	'TXL_PTET_sequestration_Kd'	, exp(-2.7)	, [exp(-10) exp(10)]	
16	'TXL_INDUCER_TETR_ATC_Kd'	, exp(-6)	, [exp(-15) exp(15)];	

activeNames(cell2mat(mter_phase2_params(:,1)),2) = mter_phase2_params(:,2);
 activeNames(cell2mat(mter_phase2_params(:,1)),3) = mter_phase2_params(:,3);
 activeNames(cell2mat(mter_phase2_params(:,1)),4) = mter_phase2_params(:,4);

estParamsIX = [1 2 7 11 16 21 23 28 31 33 35 37 39 40];
 estParams = activeNames(estParamsIX,1);

Full C.

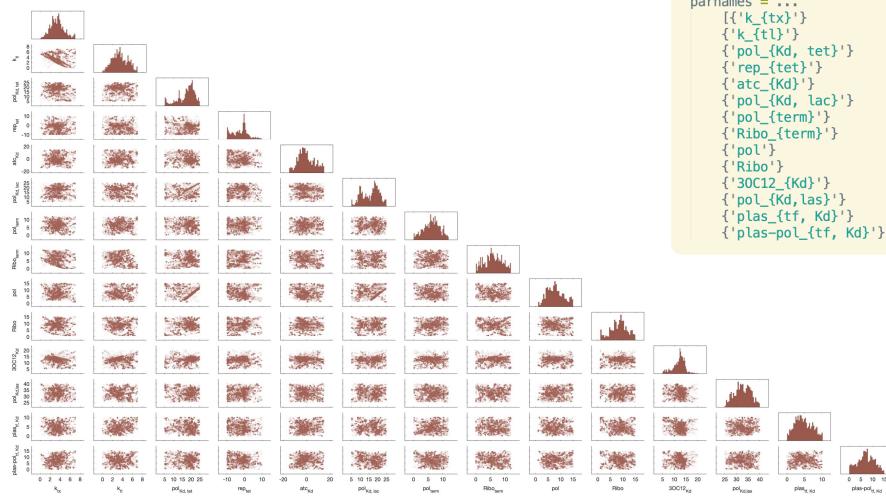
These numbers got fixed in D.

```

.5 % plot data from existing simulations.
16 tsIDhouse = 1;
17 tsIDtrous = 2;
18 switch tsIDhouse
19 case 1 % this is after including the termination parameters.
20 ts1 = '20190509_092707_1_1476';
21 ts2 = '20190509_100527_1_1478';
22 ts3 = '20190509_100552_2_1388';
23 ts4 = '20190509_100552_2_3554';
24 ts5 = '20190510_084333_1_369';
25 ts6 = '20190510_084333_1_369';
26 ts7 = '20190510_084333_1_374';
27 ts8 = '20190511_014023_1_74';
28 tstamp = [ts1 ts2 ts3 ts4 ts5 ts6 ts7 ts8];
29 niterID = 1:28 1:28 1:28 1:28 1:28 1:28 1:28 1:28;
30 load([projdir '/simdata_1 ts1 '/full_variable_set_ ts1']);
31 'm1',...
32 'mcmc_info', 'data_info', 'r1';
33 end
34 tsSave = ts8;
35 m1.masterVector;
36 marray_full = mcmc_get_walkers(tstamp,niterID, projdir);
37 marray_full = marray_full(:,1,:);
38 clear marray_full;
39 parameters = ...
40 {[('k_{\lambda}')...
41 ('k_{\alpha}')...
42 ('k_{\beta}')...
43 ('rep_{(tet)}')...
44 ('atc_{(Kd)}')...
45 ('pol_{(Kd, lac)}')...
46 ('pol_{(lac)}')...
47 ('Riboz_{term}')...
48 ('pol')...
49 ('pol_{(Kd)}')...
50 ('3OC12_{(Kd)}')...
51 ('pol_{(Kd, las)}')...
52 ('plas_{(tf, Kd)}')...
53 ('plas-pol_{(tf, Kd)}')...
54 ('plas-pol_{(tf)}')];

```

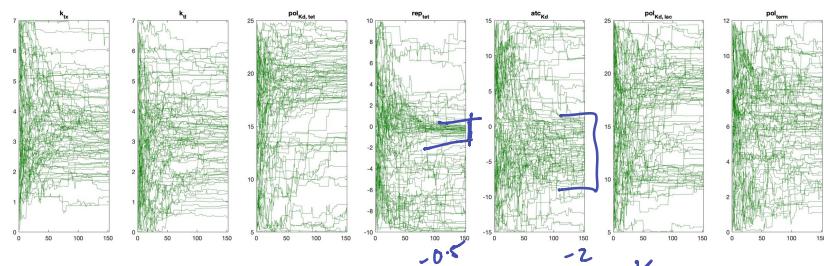
Full C



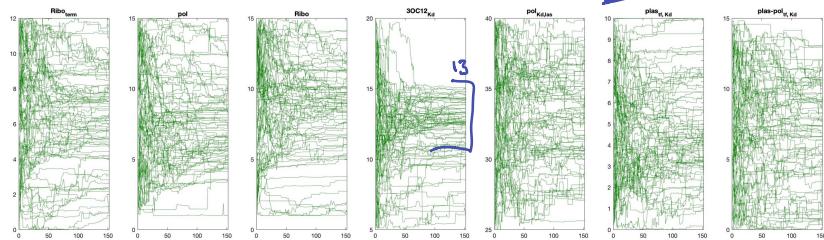
```

plotflag = true;
switch tsIDtouse
case 1 % this is after including the termination par:
    ts1 = '20190509_092707_1_1476';
    ts2 = '20190509_160552_1_1476';
    ts3 = '20190509_160552_2_738';
    ts4 = '20190509_160552_3_554';
    ts5 = '20190510_064333_1_369';
    ts6 = '20190510_064333_2_185';
    ts7 = '20190510_064333_3_74';
    ts8 = '20190511_014023_1_74';
    tstamp = [ts1 ts2 ts3 ts4 ts5 ts6 ts7 ts8];
    nIterID = [1:1:20 1:20 1:20 1:20 1:8 1:5];
    load(['projdir '/simdata_ ts1 '/full_variable_se...
        'mi',...
        'mcmc_info', 'data_info', 'ri']);
end
tsToSave = ts8;
mai.masterVector;
marray_full = mcmc_get_walkers(tstamp,nIterID, projdir);
marray = marray_full(:, :, 1:end);
clear marray_full
parnames = ...
    {'k_{tx}'}
    {'k_{ty}'}
    {'pol_{Kd, tet}'}
    {'rep_{tet}'}
    {'atc_{Kd}'}
    {'pol_{Kd, lac}'}
    {'pol_{term}'}
    {'Ribo_{term}'}
    {'pol'}
    {'Ribo'}
    {'3OC12_{Kd}'}
    {'pol_{Kd, las}'}
    {'plas_{tf, Kd}'}
    {'plas-pol_{tf, Kd}'};

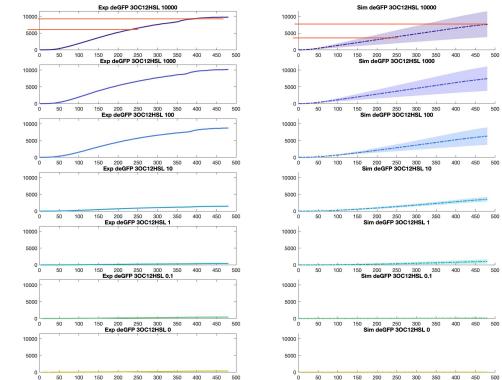
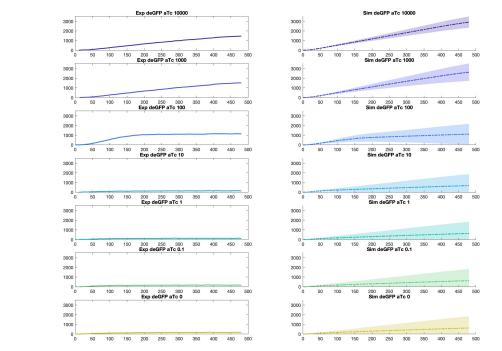
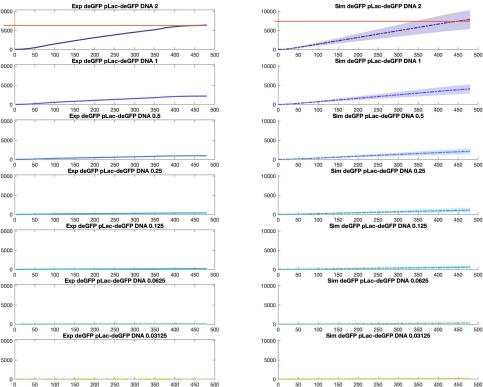
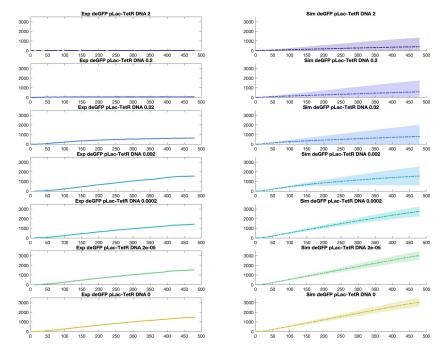
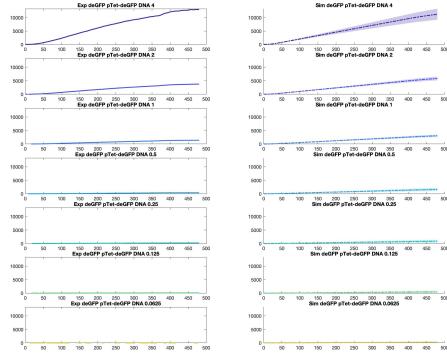
```



These three
params get
fixed in the next
iteration.



Full C hits

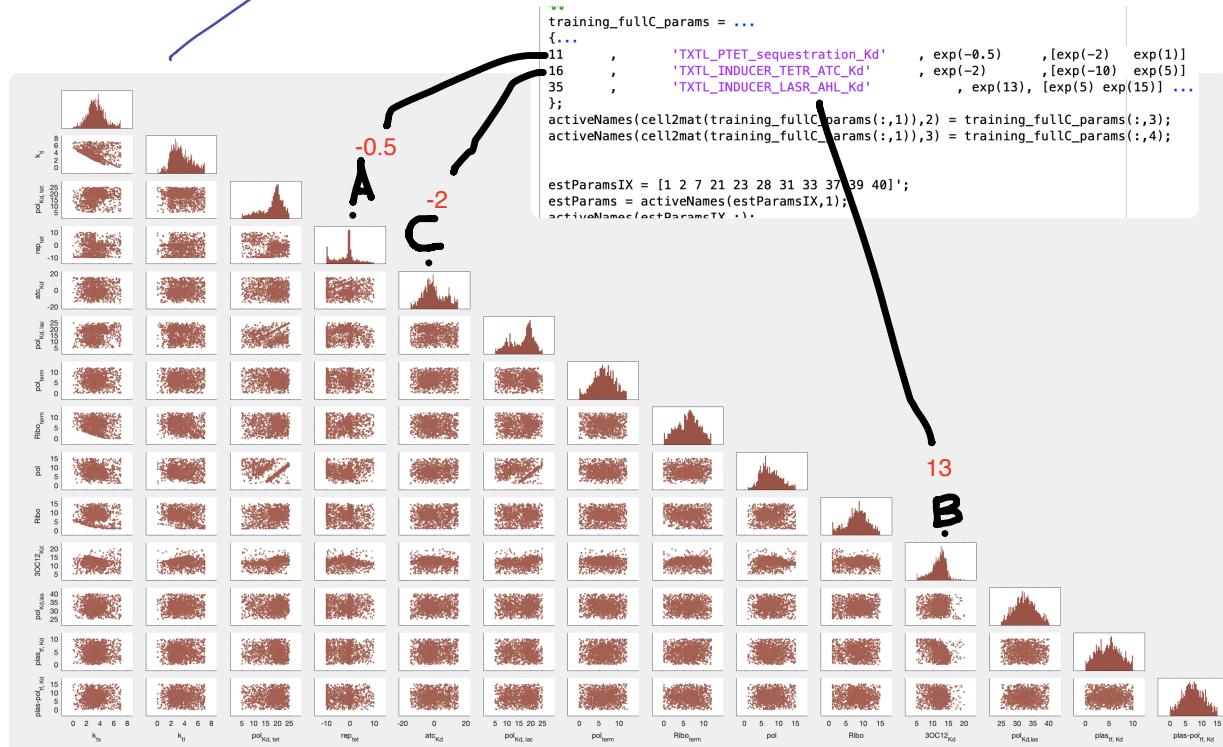


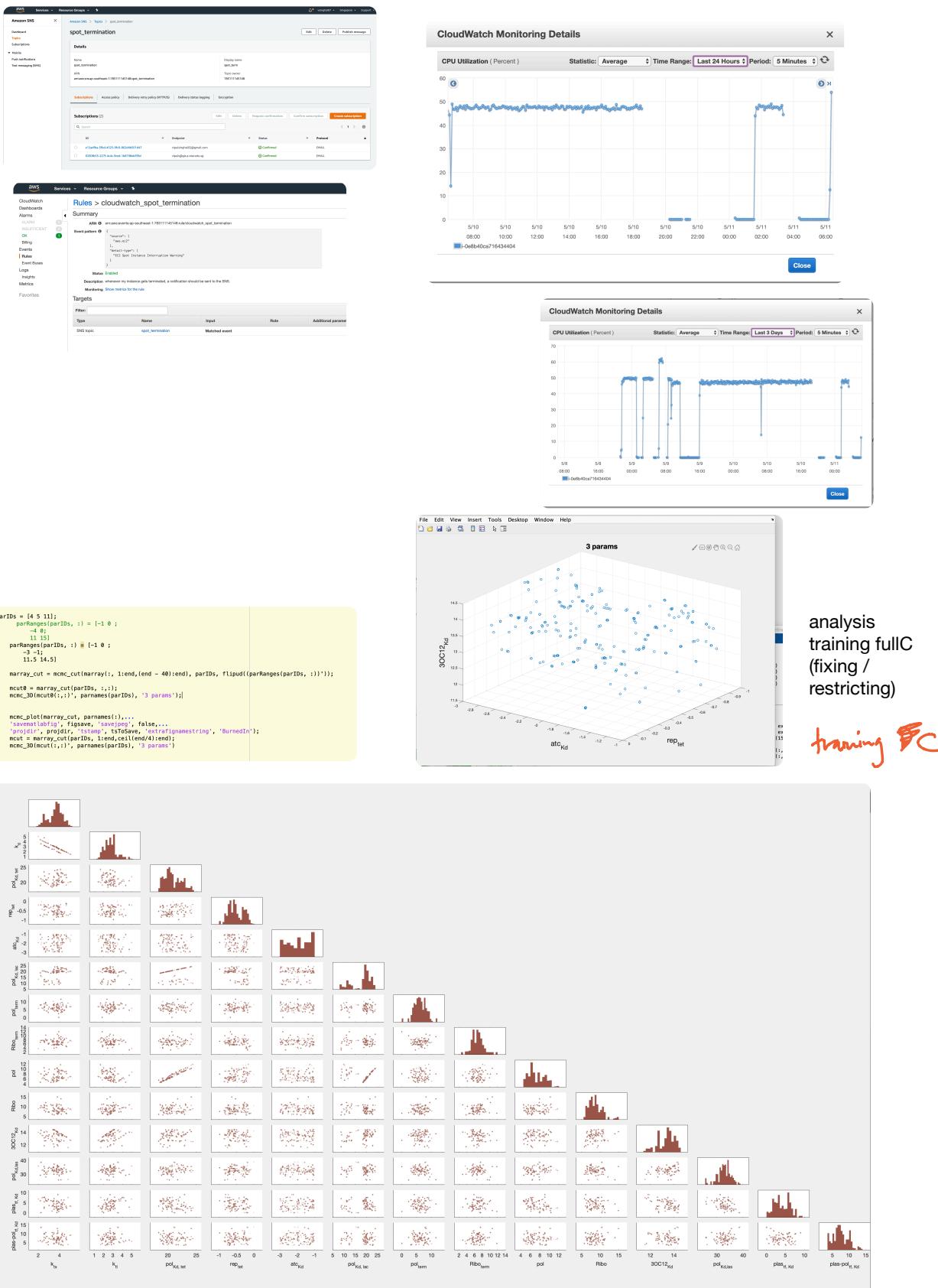
going to do the restriction procedure thing to slice out parameters, and then fix them in a consistent way. you cannot just pick parameters by looking at the plot below. if you fix parameters A and B, you have to make sure they are consistent, in the sense that there are points in the distribution that have that combination of A and B. This can be done sequentially:

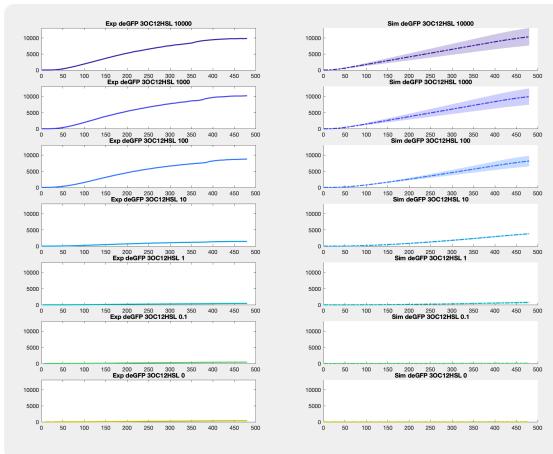
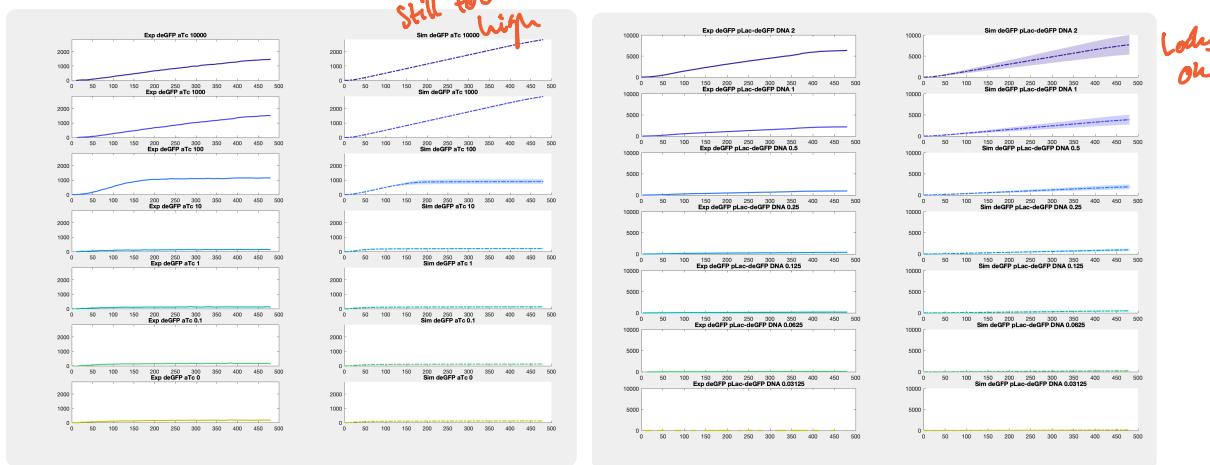
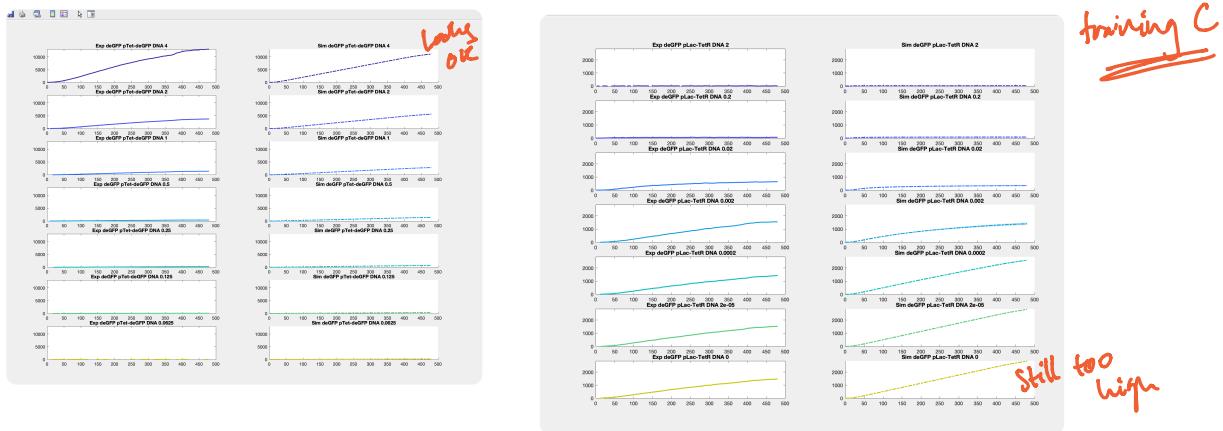
see below.

training C.

commit: [f5a61358cd9c1f4f24b5d6bd140bbce5d5f0719e](#)







At least the 3OC12 curves are high enough.

```

325
326
327
328 {
...
329   11   , 'TXL_PTEL_sequestration_Kd' , exp(-8.5) , [exp(-2) exp(1)]
330   16   , 'TXL_INDUCER_TETR_ATC_Kd' , exp(-2) , [exp(-10) exp(5)]
331   19   , 'TXL_INDUCER_LASK_AH_Kd' , exp(13) , [exp(5) exp(15)] ...
332   ;
333 - activeNames(cell2mat(training_fullC_params(:,1)),2) = training_fullC_params(:,3);
334 - activeNames(cell2mat(training_fullC_params(:,1)),3) = training_fullC_params(:,4);
335

```

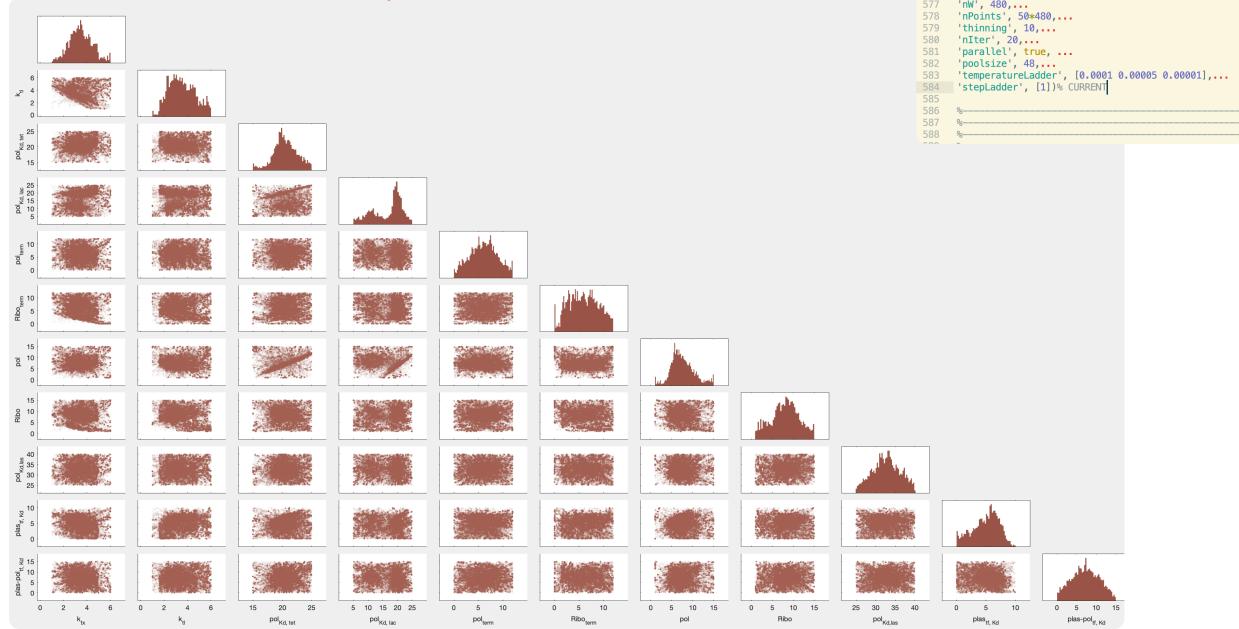
Full D-

• Fixed rep Kd (sequestration k_d), atc k_d & sociz k_d.

in fullID, I have the following. I think I am going to fix the kTX and kTL and estimate the rest. there are potentially 2 non identifiability trios: ktx, pol and polterm as 1, and ktl, ribo, riboterm as the second. Maybe the problem gets easier if I fix either pol or ktx (dont want to fix the terms I think) Actually, I want to do 3 sims, with each of these fixed, and see what happens.

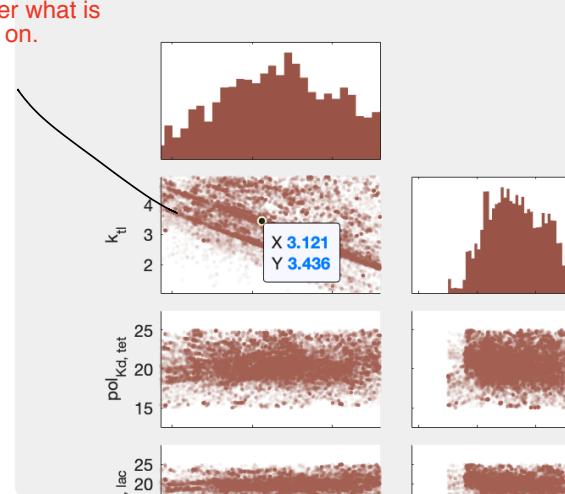
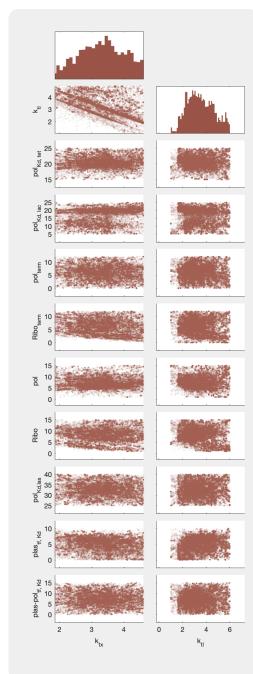
From Full C to get this. Full D.

• in Full D, going to fix k_{tx}, k_{tl} to get Full E.



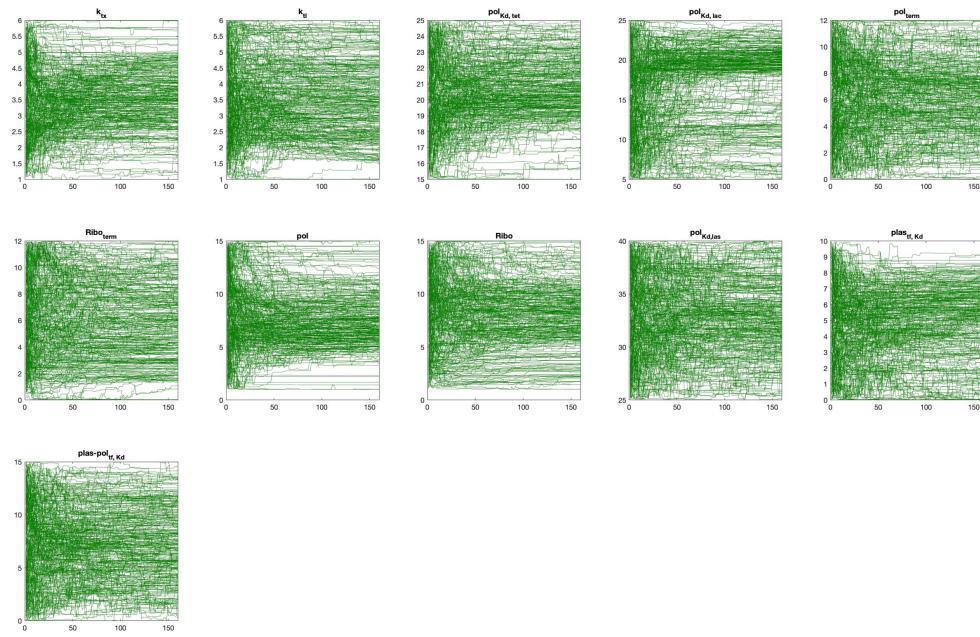
why are there two
lines! so
interesting! I
wonder what is
going on.

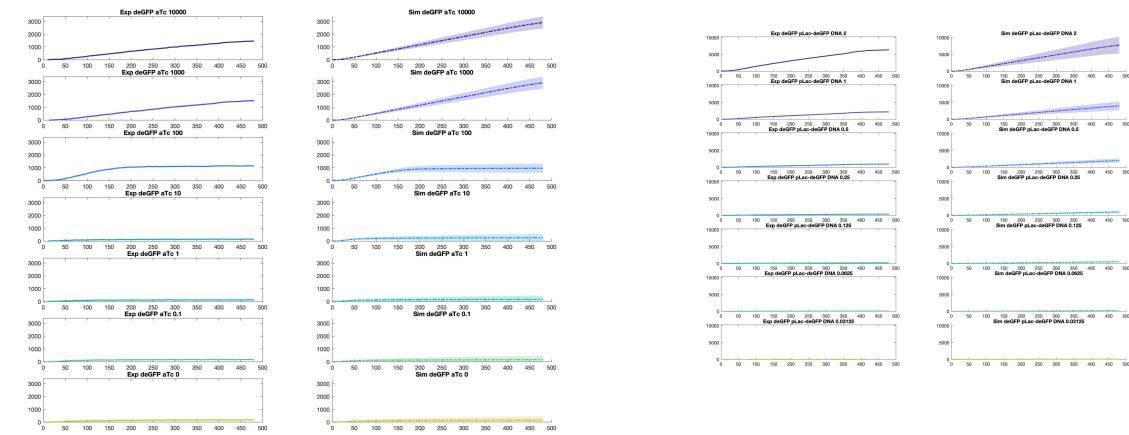
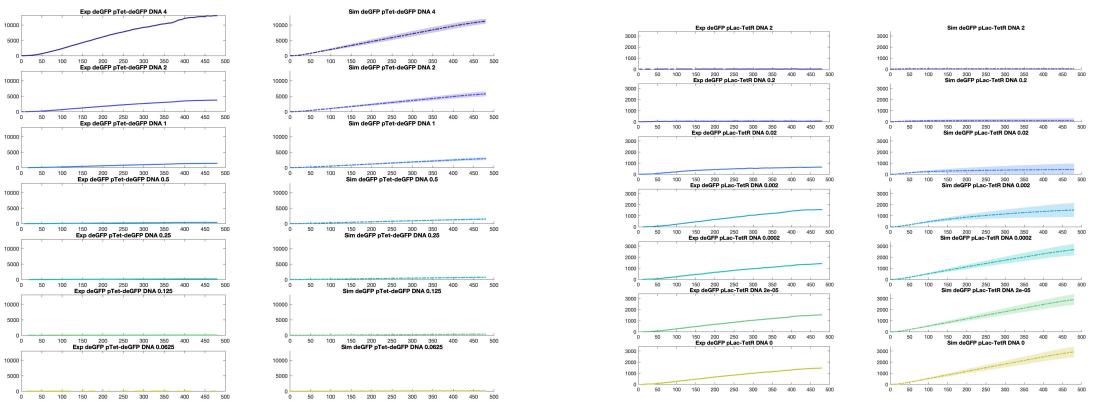
lets
fix ktx
and
ktl
first:



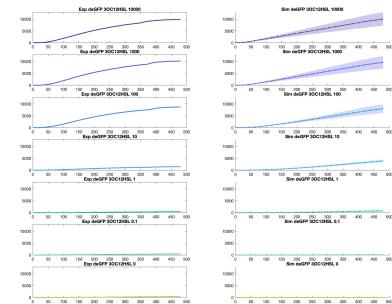
this spottiness you see
is the chains getting
stuck, because the
temperature is too low.
need acceptance of
0.23, and see the temp
in training E below, that
gets a good mixing, and
is not spotty.

talk about this
spottiness in the supp
info.

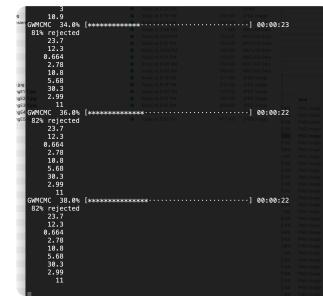




- /Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsim_vsfork2017/mcmc_simbio/projects/proj_ZSIFFL_trainingD/simdata_20190511_131947_2_74/traj20190511_131947_2_741proj_ZSIFFL_trainingD1 1.jpg



fix to these values. start training_fullE



Trainning FullE -- 84% rejection with super high temp of 1.5k (.02% of signal) Seems like the temp should be slightly higher than even this. but its ok, i will stay at this for now. 40 iters == 14 hours. started at like 11.

proj_ZSIFFL_trainingE('stepSize', 1.2,...
'nW', 480,...

```
SEARCHED [REDACTED]
GMCMC 0.0% [*****] 00:00:01
83% rejected
24.8
12.1
12.2
2.43
30.5
5.39
GMCMC 96.0% [*****] 00:00:01
84% rejected
24.8
12.1
12.2
2.43
30.5
5.39
GMCMC 98.0% [*****] 00:00:00
84% rejected
24.8
12.1
12.2
2.43
30.5
5.39
Elapsed time is 1356.065360 seconds.
Iteration number 4.
starting mcmc 4
current step size: 1.200000e+00
ending mcmc 3
```

'nPoints', 50*480,...
'thinning', 10,...
'nIter', 40,...
'parallel', true,...
'poolsize', 48,...
'temperatureLadder', [0.0002],...

'stepLadder', [1])

posterior distributions are not spotty, ie, the rejection probability is not too high. ie the chains are not getting stuck. i need rejection to be around 77%. this is still too high. going to raise temperature more, or lower step size.



/Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsim_vsfork2017/mcmc_simbio/projects/analysis_ZSIFFL_training_fullE.m

why are there 2 lines here, what is going on? so intreseting. can try to plot 3 at a time 3d Plots to try to see if you can see a trend for why there are 2 seprate lines.

also, what is going on with this knee behavior?

These sims need to run to exhaustion / till its is super clean. Then these trends will be super clear.

```

parnames = ...
[...
{'pol_{Kd,tet}'}, {'TXTL_PTET_RNAPbound_Kd'}, ]
{'pol_{Kd, lac}'}, {'TXTL_PLAC_RNAPbound_Kd'}, ]
{'pol'}, {'RNAP'}, ]
{'Ribo'}, {'Ribo'}, ]
{'3OC12_{Kd}'}, {'TXTL_INDUCER_LASR_AHL_Kd'}, ]
{'3OC12_{F}'}, {'TXTL_INDUCER_LASR_AHL_F'}, ]
{'pol_{Kd, las}'}, {'TXTL_PLAS_RNAPbound_Kd'}, ]
{'pol_{F, las}'}, {'TXTL_PLAS_RNAPbound_F'}, ]
{'plas_{tf, Kd}'}, {'TXTL_PLAS_TFBIND_Kd'}, ]
{'plas-pol_{tf, Kd}'}, {'TXTL_PLAS_TFRNAPbound_Kd'}, ]
{'plas-pol_{tf, F}'}, {'TXTL_PLAS_TFRNAPbound_F'}, ]
{'plas_{tf, F}'}, {'TXTL_PLAS_TFBIND_F'}, ]
];
% activeNames(estParamsI)

```

Reaction: '[protein lasR] + OC12HSL <-> [OC12HSL:protein lasR]'
 ReactionRate: '**TXTL_INDUCER_LASR AHL F***[protein lasR]*OC12HSL - TXTL_INDUCER_LASR_AHL_R*[OC12HSL:protein lasR]'

Reaction: '[DNA plas--utr1--deGFP] + RNAP <-> [RNAP:DNA plas--utr1--deGFP]' ~~+~~

ReactionRate: '**TXTL_PLAS_RNAPbound_F***[DNA plas--utr1--deGFP]*RNAP - TXTL_PLAS_RNAPbound_R*[RNAP:DNA plas--utr1--deGFP]' ~~+~~

Reaction: '[DNA plas--utr1--deGFP] + [OC12HSL:protein lasR] <-> [DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

ReactionRate: '**TXTL_PLAS_TFBIND_F***[DNA plas--utr1--deGFP]*[OC12HSL:protein lasR] - TXTL_PLAS_TFBIND_R*[DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

Reaction: '[DNA plas--utr1--deGFP:OC12HSL:protein lasR] + RNAP <-> [RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

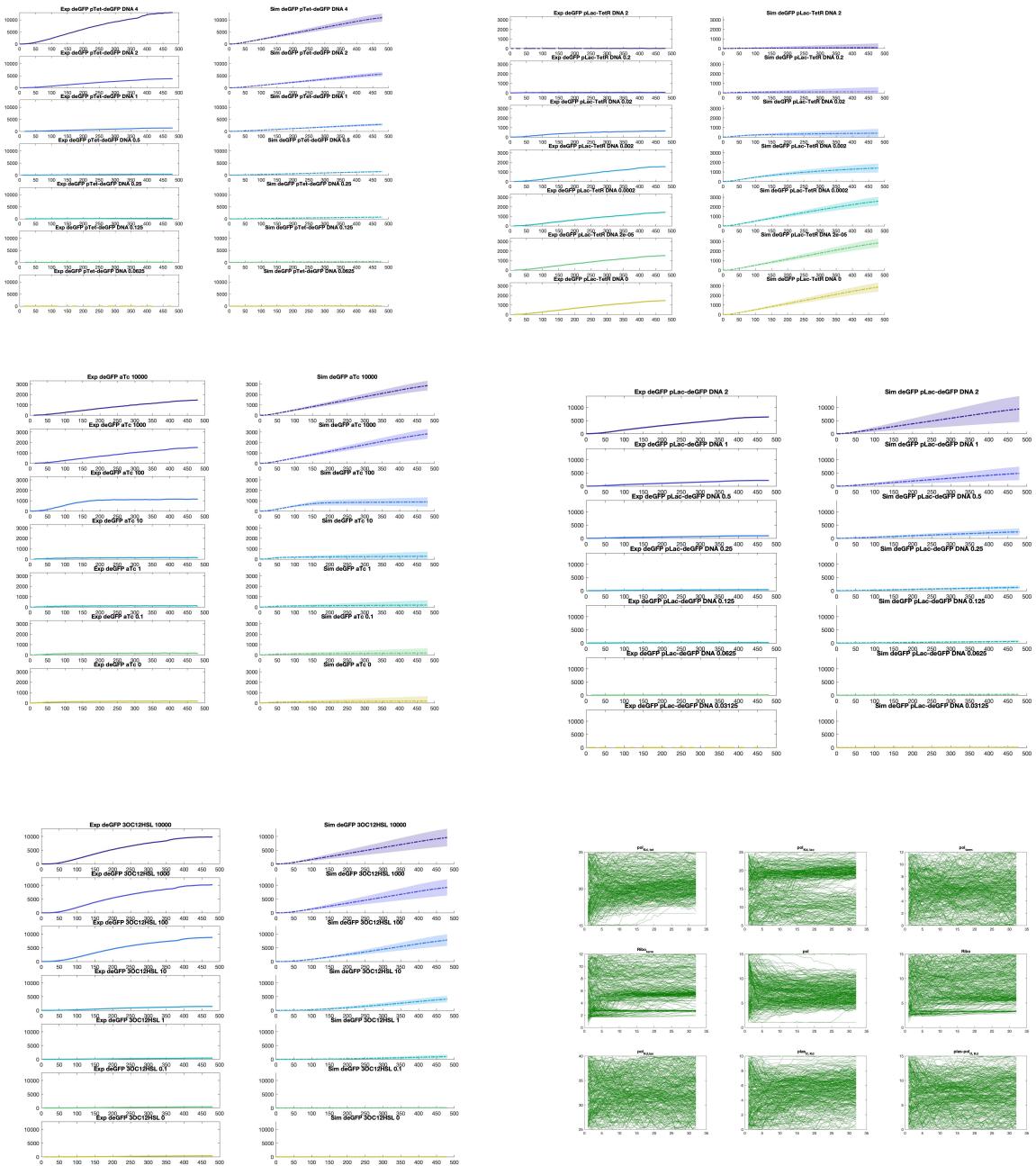
ReactionRate: '**TXTL_PLAS_TFRNAPbound_F***[DNA plas--utr1--deGFP]*[OC12HSL:protein lasR]*RNAP - TXTL_PLAS_TFRNAPbound_R*[RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

Reaction: '[RNAP:DNA plas--utr1--deGFP] + [OC12HSL:protein lasR] <-> [RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

ReactionRate: '**TXTL_PLAS_TFBIND_F***[RNAP:DNA plas--utr1--deGFP]*[OC12HSL:protein lasR] - TXTL_PLAS_TFBIND_R*[RNAP:DNA plas--utr1--deGFP:OC12HSL:protein lasR]' ~~+~~

Dont know this was ever estimated. We fixed from the moment they were introduced

WAS THIS THE
ORIGINAL
TRAINING
FULL ???



• /Users/vipulsinghal/Dropbox/Documents/toolbox/txtlsim_vsfork2017/mcmc_simbio/projects/
proj_ZSIFFL_trainingE/simdata_20190512_155207_1_1845/
trace20190512_155207_1_1845Without_transient.jpg

Seems like to some degree it is just not possible to fit to the data too perfectly. Might plot the error bars in the data too.

Write up on:

Workplan:

- predict the trajectories for the IFFL
- plot out the things listed in the other note.

ONCE THIS PAPER IS DONE, EMAIL RMM ABOUT APPLYING FOR AWS GRANTS. TELL HIM HOW MUCH I HAVE SPENT!

Can also look at applying for other grants.

June 6.

Building the IFFL with the toolbox.

- the combinatorial promoter file is pretty much all that is needed to be compatible with the way the repressor and activator promoters were written and characterized. Here I reproduce the reaction in those two promoter files and see if they agree with the combinatorial promoter file.

plas promoter file



TF: OC12:HSL: protein lasR

- TXTL_PLAS_RNAPbound_F/R
- TXTL_PLAS_TFBIND_F?R
- TXTL_PLAS_TFRNAPbound_F/R
- TXTL_PLAS_TFBIND_F/R

June 8, 2019: As it stands: the combinatorial promoter file lacks the leaky expression reaction: RNAP binding to the unactivated promoter, with the reaction rate parameter:

TXTL_PLAS_RNAPbound_F/R

Since we estimated that, I will be adding that to the combinatorial promoter file right now. its value estimated above was around 35 on the log e scale.

ptet promoter file



TXTL_PTET_RNAPbound_F/R
TXTL_PTET_sequestration_F/R

plas-ptet combinatorial file

repression surpasses activation.

- TF = OC12:lasR
- activ_bound = RNAP:DNA:OC12:lasR or DNA:OC12:lasR
- tetRdimer

---- activation and RNAP binding ----



DNA:TF + RNAP \leftrightarrow DNA:TF:RNAP == RNAPbound:TF (this is what proceeds to transcription)



---- REPRESSION: tetR dimer binding DNA ----



---- REPRESSION: tetR binding to activated DNA, and popping off activator (and RNAP if present) ----

RNAP:DNA:OC12:lasR + tetRdimer



$\leftrightarrow \text{RNAP} + \text{DNA:tetRdimer} + \text{OC12:lasR} [[\text{TXTL_COMBINATORIAL_ACTIVATOR_KNOCKOFF_F/R}]] == 100x$ more dissociate-y than the dna-tf binding. $\Rightarrow [[\text{TXTL_PLAS_TFBIND_100xF/R}]]$ -- note that the FORWARD reaction is the dissociation here! -- so the R sets the timescale. so need to vary Kd and F together when trying the range of values.

DNA:OC12:lasR + tetRdimer



$\leftrightarrow \text{DNA:tetRdimer} + \text{OC12:lasR} [[\text{TXTL_COMBINATORIAL_ACTIVATOR_KNOCKOFF_F/R}]] == 100x$ more dissociate-y than the dna-tf binding. $\Rightarrow [[\text{TXTL_PLAS_TFBIND_100xF/R}]]$

Just do a study showing that it is insensitive to the value picked: try 10x to 1000x. -- basically once tetR dimer binds, it almost certainly makes the activator fall off.

```

get(mIFFL.Reactions(79))
  Active: 1
  Annotation: "
    Name: "
    Notes: "
    Parent: [1x1 SimBiology.Model]
    Products: [3x1 SimBiology.Species]
    Reactants: [1x1 SimBiology.Species]

    Reaction: [RNAP:DNA plas_ptet--utr1--deGFP:OC12HSL:protein lasR:protein tetRdimer] <-> [DNA plas_ptet--utr1--deGFP:protein tetRdimer] + RNAP + [OC12HSL:protein lasR]

    ReactionRate: 'TXTL_COMBINATORIAL_ACTIVATOR_KNOCKOFF_F*[RNAP:DNA plas_ptet--utr1--deGFP:OC12HSL:protein lasR:protein tetRdimer] - TXTL_COMBINATORIAL_ACTIVATOR_KNOCKOFF_R*[DNA plas_ptet--utr1--deGFP:protein tetRdimer]*RNAP*[OC12HSL:protein lasR]'

    Reversible: 1
    Stoichiometry: [-1 1 1]
    KineticLaw: [1x1 SimBiology.KineticLaw]
      Tag: "
        Type: 'reaction'
      UserData: []

```



```

>> get(mIFFL.Reactions(65))
  Active: 1
  Annotation: "
    Name: "
    Notes: "
    Parent: [1x1 SimBiology.Model]
    Products: [1x1 SimBiology.Species]
    Reactants: [2x1 SimBiology.Species]

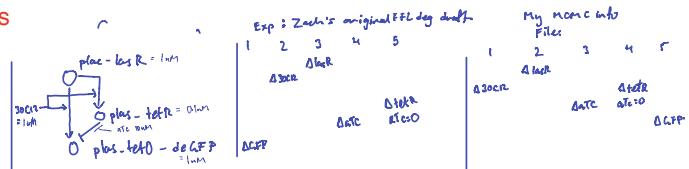
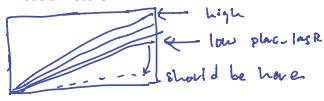
    Reaction: '[DNA plas_ptet--utr1--deGFP] + [OC12HSL:protein lasR] <-> [DNA plas_ptet--utr1--deGFP:OC12HSL:protein lasR]'

    ReactionRate: 'TXTL_PLAS_TFBIND_F*[DNA plas_ptet--utr1--deGFP]*[OC12HSL:protein lasR] - TXTL_PLAS_TFBIND_R*[DNA plas_ptet--utr1--deGFP:OC12HSL:protein lasR]'

    Reversible: 1
    Stoichiometry: [-1 -1 1]
    KineticLaw: [1x1 SimBiology.KineticLaw]
      Tag: "
        Type: 'reaction'
      UserData: []

```

- Somewhere there is a bug in the plas activation.



- SOC IR induction is working OK

- Las IR activation is too leaky.

-

- check lasr levels in 2
- check 0 lasr
-

