

UCLA

UCLA Electronic Theses and Dissertations

Title

Modeling the dynamics of the P53 regulation network with the negative feedback regulator Wip1 and P53 protein form expansion

Permalink

<https://escholarship.org/uc/item/65h4t06v>

Author

Lazrak, Amina

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Modeling the dynamics of the P53 regulation network with the
negative feedback regulator Wip1 and P53 protein form expansion

A thesis submitted in partial satisfaction
of the requirements for the degree Master of Science
in Computer Science

by

Amina Lazrak

2022

© Copyright by

Amina Lazrak

2022

ABSTRACT OF THE THESIS

Modeling the dynamics of the P53 regulation network with the
negative feedback regulator Wip1 and P53 protein form expansion

by

Amina Lazrak

Master of Science in Computer Science

University of California, Los Angeles, 2022

Professor Joseph DiStefano III, Chair

In response to cellular stress, such as DNA damage, the tumor suppressor p53 is activated to regulate hundreds of genes involved in DNA damage repair, cell cycle arrest, apoptosis, and senescence. The regulation of p53 activation by Hdm2, HdmX, and kinases such as ATM, results in different dynamics in cancerous cells and normally proliferating cells. Multiple models have been published to quantitatively capture the dynamics of this regulation and have included between 2 and 20 compartments. We developed 5 and 6-compartment mathematical models of P53 regulation using ordinary differential equations (ODEs) by incorporating Wip1 mRNA and protein, a molecular gatekeeper of the p53 autoregulatory loop. We performed parameter estimation formulated as an optimization problem on noncancerous cell data to determine the values of 23 parameters and achieved a best fit for the 5-state variable model using a combination of global and local search algorithms and maximizing a log likelihood cost function. A couple of measured data points were not fitted within a 20% confidence interval and multiple parameters sets showed correlation through a covariance matrix analysis. We therefore explored the expansion of the P53 protein into its unphosphorylated and phosphorylated form as another method of capturing the complexity of the model and achieved a best fit using the Simplex search algorithm and a nonlinear least squares cost function where all measured data points were fitted within a 20% confidence interval. We tested this model on additional cancerous data yet only achieved an adequate fit with relatively large residuals and we propose additional tools such as parameter regularization and fitting on simulated data to achieve a more robust and accurate model.

The thesis of Amina Lazrak is approved.

Sriram Sankararam

Quanquan Gu

Joseph DiStefano III, Committee Chair

University of California, Los Angeles

2022

Table of Contents

INTRODUCTION	1
P53 REGULATORY NETWORK	1
MODELING P53 DYNAMICS.....	3
MODEL DESCRIPTIONS	5
CANDIDATE MODELS FOR P53 REGULATORY DYNAMICS	5
INPUT AND OUTPUT DATA	8
SOFTWARE TOOLS	9
MATLAB 2019B/2021A.....	9
SIMULINK.....	9
AMIGO2	10
PARAMETER ESTIMATION METHODS.....	11
OPTIMIZATION PROBLEM	11
CORRELATION MATRICES	12
SENSITIVITY ANALYSIS.....	12
RESULTS.....	13
BEST FIT FOR WIP1 MODEL	13
BEST FIT FOR P53 EXPANSION.....	17
DISCUSSION	21
APPENDIX A	24
APPENDIX B	25
APPENDIX C	27
BIBLIOGRAPHY	35

Introduction

P53, often referred to as “the guardian of the genome”, acts as a tumor suppressor that is activated in response to cellular stress including DNA damage, hypoxia, oncogene activation and ribosomal stress [1]. This gene prevents tumorigenesis by either preventing the accumulation of oncogenic mutations or limiting the consequences of an oncogenic mutation that would otherwise allow uncontrolled proliferation [2]. The importance of these functions implies that P53 itself must be carefully regulated by a feedback mechanism for the normal proliferation of cells.

P53 Regulatory network

Following a cellular stressor such as a double-stranded DNA break (DSB), p53 is activated through a process by multiple molecules [1]. First, the Mre11-Rad50-Nbs1 (MRN) complex detects these breaks and recruits inactive ATM dimers to the site of the DSBs, activating ATM through auto-phosphorylation [9]. This phosphorylation cascade results in signal amplification leading to the phosphorylation of p53 at multiple serine sites including Serine 15 on its N terminus which stabilizes the p53 molecule [9]. ATM also phosphorylates other proteins such as checkpoint kinase (Chk2) which in turn adds to the phosphorylation of p53 at Serine 20 [10]. These Serine sites serve as the binding site for Hdm2 on P53 and the phosphorylation at these sites by ATM prevents Hdm2 from degrading p53 [1]. ATM and Chk2 also phosphorylate Hdm2 and HdmX to facilitate their degradation by Hdm2 polyubiquitination [5]. Therefore, removing these negative regulators of p53 and stabilizing the protein itself results in a rapid increase in total p53 protein concentration [3].

The regulatory network of p53 includes Hdm2, HdmX, and Wip1 [3]. Hdm2 is an E3 ubiquitin ligase that negatively regulates p53 transcription factor levels. It forms homo-oligomers to polyubiquitylate p53 protein resulting in degradation by proteasomes [4]. HdmX is a non-redundant p53 regulator that heterodimerizes with Hdm2 to form a more effective p53 E3 ubiquitin [5]. Both Hdm2 and HdmX can also inhibit p53 transactivation by using their N-terminal hydrophobic pockets to occupy the p53 protein's amino-terminal transactivation domain [5]. This prevents p53 from interacting with coactivators that acetylate p53 to enable it to activate G1 cell cycle arrest [3, 6, 7]. Yet depending on the levels of HdmX, it can also act as a positive feedback loop as the inhibitory forward paths from the kinases to HdmX to hdm2 mRNA results in a net positive feed forward effect on Hdm2 [8].

Wip1 serine/threonine phosphatase has garnered attention in p53 regulatory networks as its overexpression inhibits p53 functions in several tumor types [12]. Wip1 has both direct and indirect roles in inhibiting p53 and serves as a third negative feedback loop in the system. Wip1 directly destabilizes p53 by binding to the Serine 15 site and dephosphorylating the protein [12]. Wip1 exerts its indirect effects by acting on the kinases ATM and Chk2 as well as Hdm2 and HdmX. Wip1 dephosphorylates and inactivates ATM and Chk2 kinases, preventing them from phosphorylating and stabilizing p53. Wip1 also prevents ATM and Chk2 from inactivating the negative regulators of the p53 regulation network [13]. Simultaneously, Wip1 dephosphorylates Hdm2 and Hdmx at the same serines where ATM and Chk2 phosphorylate them which directly stabilizes Hdm2 and Hdmx and prevents Hdm2 from polyubiquitinating them for proteasomal degradation [14].

Modeling p53 dynamics

The p53 DNA damage response to DSBs in single cells has been shown to oscillate as a series of discrete pulse with a period of 4-7 hours [15]. These oscillations are the result of the combination of the p53-Hdm2-HdmX positive and negative feedback loops and the repeated initiation of ATM as a result of persistent DNA damage [16, 17]. It has also been suggested that p53 pulses are excitable indicating that if a stimulus is large enough to surpass the p53 activation threshold, the impulse occurs irrespective of a persistent damage signal [17]. Recent work suggests that Wip1 is responsible for setting these cell-specific thresholds for p53 activation [18]. Outside of these conditions, P53 protein remain low and barely detectable for unstressed cells [19].

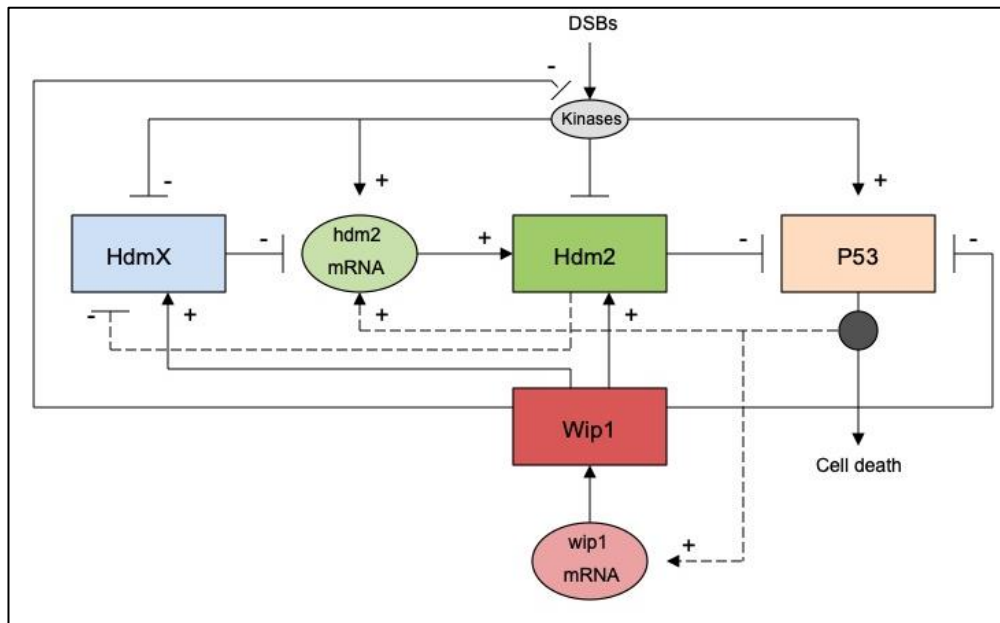
One of the most comprehensive models that simulates p53 dynamics is published by Chong, et al. in 2015 [3]. It consists of 19 ordinary differential equations (ODEs) capturing the dynamics of mRNA and protein levels for p21 phosphorylated ATM, Mdm2 and MdmX as well as their phosphorylated versions, p53, doubly phosphorylated p53, acetylated p53, and the complexes p53-Mdm2, p53-Mdmx, and Mdm2-Mdmx [3]. Although this model is successful in encompassing a large amount of key players in the regulation of p53, its complexity makes it very difficult to gather enough experimental data to produce a structurally identifiable model. As a result, the Chong et al. model is trained using a combination of data from literature and simulated data. In contrast, we developed a p53 regulatory network model based on biological data and recorded values for p53, Hdm2, HdmX and Hdm2 mRNA in addition to a measure of p53 serine-15 phosphorylation as a surrogate for ATM activity. Therefore, we extend a 4-state variable model to include either Wip1, Wip1 mRNA or expanding P53 into multiple forms with to fit normally proliferating WS1 skin cells [20]. The input for this model is serine-15

phosphorylation levels over time and represents the net effect of phosphorylation and dephosphorylation at this epitope for total P53.

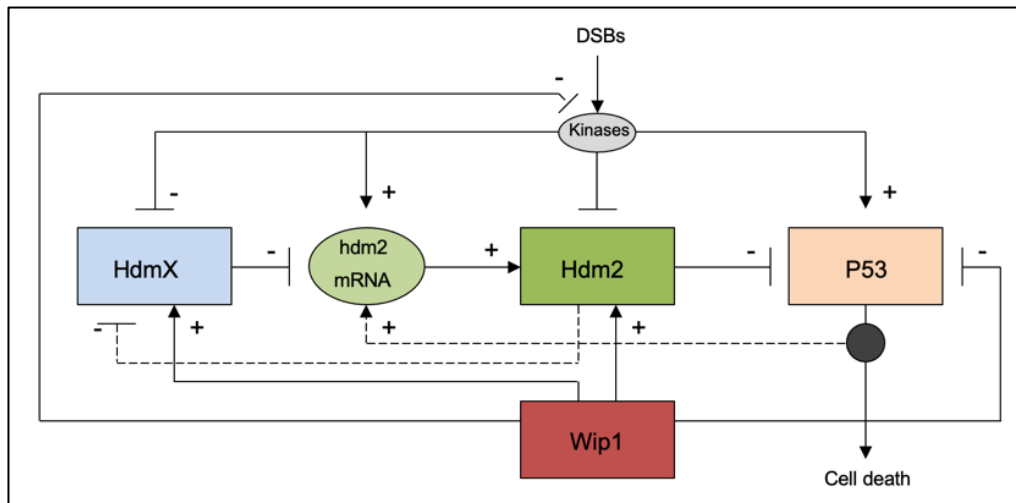
Model Descriptions

Candidate models for p53 regulatory dynamics

p53 regulation has been modeled by many research groups and mathematical models have ranged between 2 to 20 state variable models. We extended the model from Sin 2012 from a 4-compartment model to a 5- and 6-compartment model to include the negative regulator protein Wip1 and its mRNA [20].



Compartmental model diagram of p53 regulation with Wip1 mRNA and Wip1 protein (6 state variables)



Compartmental model diagram of p53 regulation with Wip1 protein (5 state variables)

These more complex models incorporate the Wip1 protein denoted as x_5 and wip1 mRNA denoted as x_6 in addition to the previously defined state variables Hdm2 (x_1), HdmX (x_2), p53 (x_3), hdm2 mRNA (x_4). From the block diagram, the following ODEs are defined:

$$\frac{dx_1}{dt} = p_1 x_4 (t - \tau_1) - p_3 x_1 - p_4 \left(\frac{x_1^2}{p_5 + x_1} \left(1 + \frac{p_6 u}{p_7 + u} \right) \right) + k_{15}' x_5 x_1 \quad (1)$$

where $k_{15}' = k_{15} x_5(\max)$

$$\frac{dx_2}{dt} = p_8 - p_9 x_2 - p_{10} \left(\frac{x_1 x_2}{p_{11} + x_2} \left(1 + \frac{p_{12} u}{p_{13} + u} \right) \right) \quad (2)$$

$$\frac{dx_3}{dt} = p_{14} - p_{15} x_3 - p_{16} \frac{x_1 x_3}{p_{17} + x_3} (1 - p_{18} u) - k_{35}' x_5 x_3 \quad (3)$$

where $k_{35}' = k_{35} x_5(\max)$

$$\frac{dx_4}{dt} = p_{19} - p_{20} x_4 + \frac{p_{21} x_3^4}{p_{22}^4 + x_3^4} (1 + p_{23} u) (1 - p_{24} x_1) (1 - p_{25} x_2) \quad (4)$$

For the 5 state variable model, the following equations are used:

$$\frac{dx_5'}{dt} = p_{26}' - p_{27} x_5' \quad (5)$$

where $p_{26}' = \frac{p_{26}}{x_5(\max)}$ and $x_5' = \frac{x_5}{x_5(\max)}$

For the 6 state variable model, the following equations are used:

$$\frac{dx_5'}{dt} = p_{26}'' x_6' (t - \tau_2) - p_{27} x_5' \quad (6)$$

where $p_{26}'' = \frac{p_{26} x_6(0)}{x_5(\max)}$ and $x_5' = \frac{x_5}{x_5(\max)}$

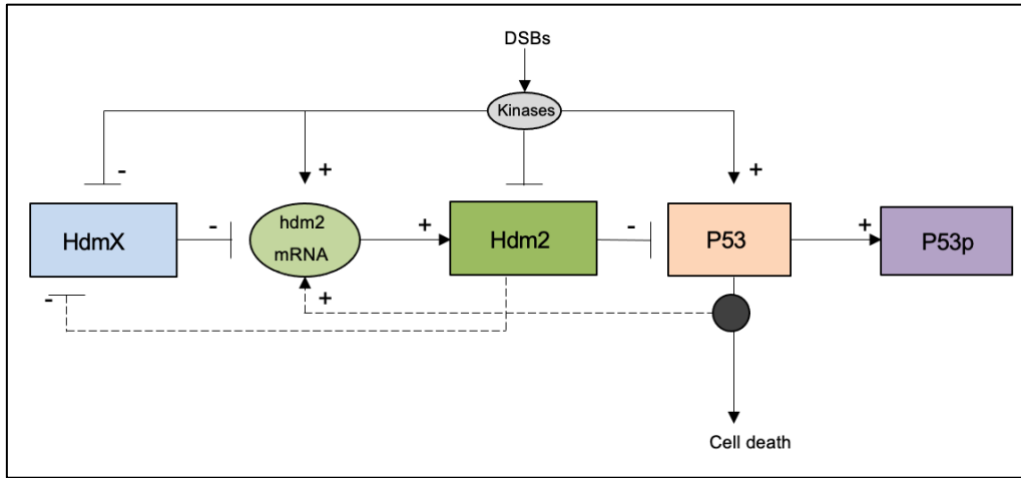
$$\frac{dx_6'}{dt} = p_{28}' + p_{29}' \left(\frac{x_3 (t - \tau_2)^4}{p_{30}^4 + x_3 (t - \tau_2)^4} \right) - p_{31} x_6' \quad (7)$$

where $p_{28}' = \frac{p_{28}}{x_6(0)}$, $p_{29}' = \frac{p_{29}}{x_6(0)}$ and $x_6' = \frac{x_6}{x_6(0)}$

The equations for Hdm2, HdmX, P53 and hdm2 mRNA are normalized to initial conditions measured experimentally and the equations for wip1 mRNA and Wip1 protein are normalized to

maximum values according to values found in the literature. Therefore, all state variables have an initial condition of 1.

We also explored an alternative model that does not include the negative regulator Wip1 but instead the expansion of the P53 protein into its unphosphorylated and phosphorylated forms, resulting in a 5 state variable model.



Compartmental model diagram of p53 regulation with phosphorylated P53 protein (5 state variables)

The phosphorylated P53 protein (P53p) is represented by x_5 added to the previously defined state variables Hdm2 (x_1), HdmX (x_2), P53 (x_3), hdm2 mRNA (x_4) in the following set of ODEs:

$$\frac{dx_1}{dt} = p_1x_4 - p_3x_1 - p_4x_1u - p_5x_1x_2 \quad (8)$$

$$\frac{dx_2}{dt} = p_8 - p_9x_2 - p_{10}x_2u - p_{11}x_1x_2 \quad (9)$$

$$\frac{dx_3}{dt} = p_{14} - p_{15}x_3 - p_{16}x_1x_3 - p_{33}x_2x_3 + p_{30}x_5 - p_{34}ux_3 \quad (10)$$

$$\frac{dx_4}{dt} = p_{19} - p_{20}x_4 + \frac{p_{21}x_5^4}{p_{22}^4 + x_5^4} \quad (11)$$

$$\frac{dx_5}{dt} = p_{34}x_3u - p_{30}x_5 \quad (12)$$

Input and Output Data

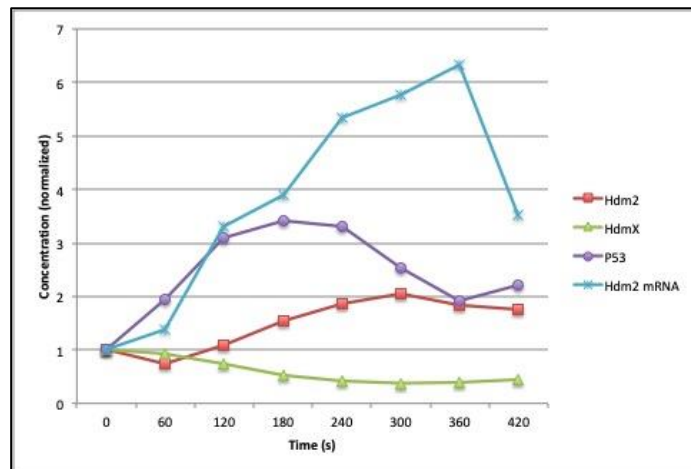
The input (u) to the six-compartment model is a fitted polynomial of 6th degree. It is fitted to a time course experiment of the phosphorylation of serine 15 on WS1 cells with an R-squared value of 0.99 and is used as an indirect measure of kinase signaling upstream of p53 activation.

Time (s)	Hdm2	HdmX	P53	Hdm2 mRNA
0	1.00	1.00	1.00	1.00
60	0.738	0.939	1.95	1.37
120	1.08	0.748	3.11	3.30
180	1.55	0.517	3.41	3.90
240	1.86	0.413	3.30	5.33
300	2.06	0.377	2.55	5.78
360	1.83	0.381	1.92	6.32
420	1.76	0.447	2.20	3.53

Model input data for WS1 cell phosphorylation

Data was collected and reported as normalized concentrations of the 4 state variables Hdm2, HdmX, p53 and Hdm2 mRNA for an 8 hour experiment in normally proliferating WS1 cells.

Time (s)	nSerine WS1
0	0.005
60	0.629
120	0.667
180	0.758
240	0.949
300	1.00
360	0.776
420	0.573



Model output data for WS1 cells

Software Tools

To determine the parameters of the candidate models and compare their fits, we utilized MATLAB and its Simulink package as well as AMIGO2 for simulation, parameter estimation and sensitivity analyses.

MATLAB 2019b/2021a

MATLAB (MATrix LABoratory) is a comprehensive environment for computing, programming and visualizing. It provides both numeric and symbolic computing abilities allowing for intensive computations using a large number of ordinary differential equations as well as simulations of complex biosystems models. MATLAB combines a desktop environment specialized for iterative analysis and allows for the extension of its functions through toolboxes including Simulink and AMIGO2, which we both used for parameter estimation and curve-fitting [22].

Simulink

Simulink is a MATLAB app that provides a graphical user interface for modeling and simulating dynamical systems. It allows for the designing of block diagrams of multi-compartmental models and offers built in analyses for these models including parameter estimation, model optimization, sensitivity analysis and model validation [24]. Due to its graphical interface, Simulink does not supply much flexibility for the user and presents challenges in extracting model information outside of its built-in tools. Contrastingly, it supplies a highly user-friendly modeling environment that allows for rapid setup for simulation and comparison of optimization algorithms and the resulting model fit. An example block diagram designed in Simulink is displayed in APPENDIX C.

AMIGO2

AMIGO2 (Advanced Model Identification using Global Optimization) is a multi-platform MATLAB toolbox developed by Eva Balsa-Canto and Julio R. Banga in 2011 with the purpose of streamlining biosystems modeling through MATLAB scripts. AMIGO2 is freely available to academic users and supplies multiple biological model examples to increase its usability [25]. We utilized AMIGO2 for model parameter identification using a metaheuristic that integrates local and global search methods and provides sensitivity analyses for the estimated parameters as well as a Hessian matrix to determine quantitatively the quality of the model fit. The AMIGO2 programming code utilized for the p53 model can be found in APPENDIX C.

Parameter Estimation Methods

We followed a systematic approach to design compartmental models through a set of ordinary differential equations, perform parameter estimation, analyze the best-fit, reduce the model and identify the importance of each parameter.

Optimization Problem

The Simulink software formulates the parameter estimation problem as an optimization problem where the solution is the set of estimated parameter values [26]. Within this optimization problem, we defined the model parameters as the design variables x and the objective function/cost function $F(x)$ to determine the difference between simulated and measured responses. We utilized the residual sum squared error (rss) for the cost function in Simulink, defined as

$$rss(x) = \sum_{t=0}^{t_N} e(t) e^T(t) \quad (13)$$

where N is 32, the number of samples, and the log likelihood (llk) cost function defined in AMIGO2 defined as

$$llk(\theta) = \sum_{i=1}^n \ln f_i(y_i|\theta) \quad (14)$$

where θ represents the parameters estimated. We also defined the bounds of the estimated parameters as $0 < x < \infty$. The goal of the optimization problem is to minimize the sum squared error cost function or maximize the log likelihood cost function and we utilized multiple optimization methods with varying levels of success. In conjunction with Simulink, we used the Simplex algorithm based on the Nelder-Mead algorithm that does not depend on the differentiability of the cost function resulting in a higher likelihood of convergence under defined criteria ranging from $1e^{-3}$

to $1e^{-5}$. We also utilized the nonlinear least squares optimization method which minimizes the squares of the residuals between the simulated and measured curves, yet this method requires a vector of error residuals calculated at each iteration resulting in a greater computational cost [23].

The optimization algorithms utilized with AMIGO2 are a combination of local and global algorithms including nonlinear least squares and ‘fminsearch’ and alternates between the algorithms when close to convergence [25].

Correlation Matrices

To determine the identifiability of individual parameters within the parameter set we derived the correlation matrix. This symmetrical matrix based on calculating the inverse of the Fisher information matrix determines the covariance between each parameter and every other parameter and indicates a poor model design if multiple sets of parameters are highly correlated [27].

Sensitivity Analysis

Another valuable analysis for the best-fit of a model is parameter ranking or “parameter sensitivity”, which employs the normalized parameter sensitivities and quantitatively describes the effect of changing a parameter on the outputs of the model. Perturbations to a parameter that result in a larger effect on the fitting output denotes the model is more sensitive to this parameter and suggests the parameter plays a role in driving the fitting process of the designed model to the biological data. For a system with state variable x_i and parameter θ_j , the relative sensitivity of x_i to θ_j is defined as

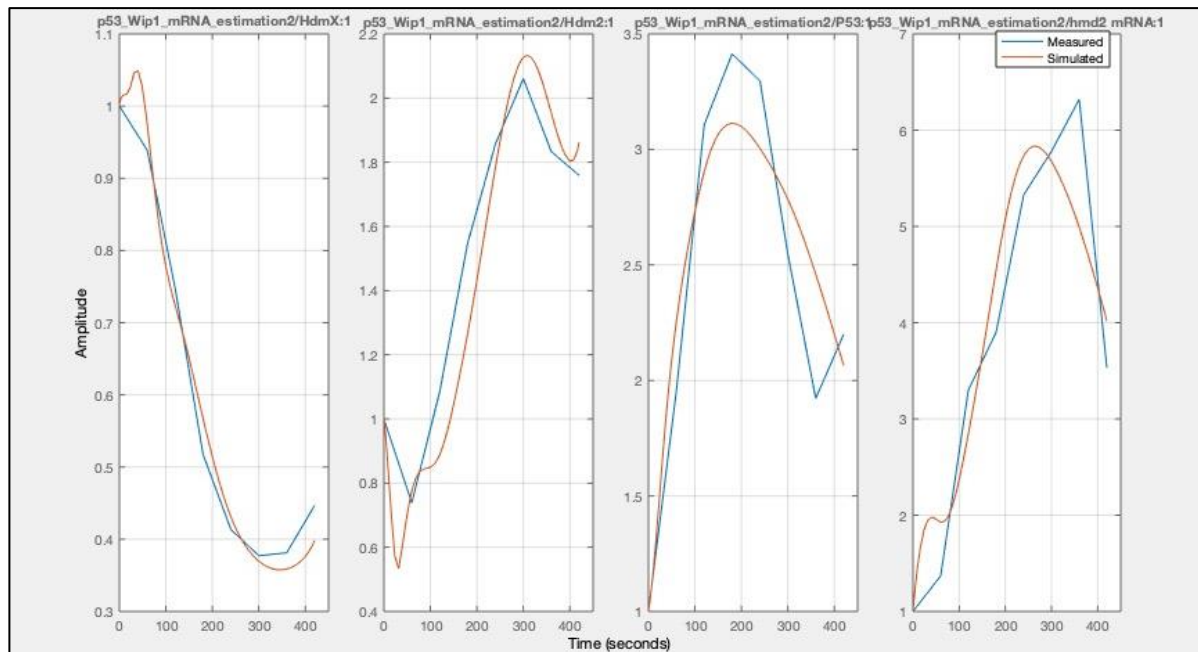
$$s_{ij}(t) = \frac{\frac{\partial x_i(t)}{\partial \theta_j}}{\frac{x_i}{\theta_j}} \quad (15)$$

[28, 29].

Results

Best Fit for Wip1 Model

We first modeled the 6 state variable dynamical model including Wip1 and wip1 mRNA using the parameter estimation tool of Simulink to estimate values for 31 parameters with the goal of fitting the WS1 cell data. This model includes a time delay fixed to 10 minutes as is reported in the literature and the starting values for the parameter search are derived from the Sin 2012 model and the Chong et al. 2015 model without any modifications [3, 20]. After running both the Simplex algorithm and the Levenberg-Marquardt algorithm to minimize the sum of squared errors with the minimal function tolerance of $1e^{-3}$, we obtained the following fit for the measured data.

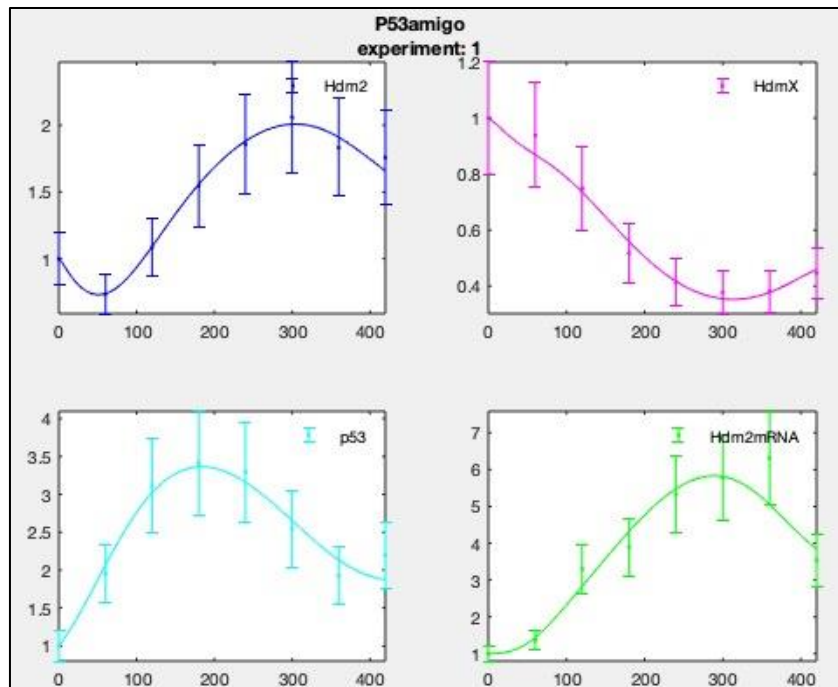


Parameter estimation results of 6-state variable model with Wip1 and Wip1 protein fitting the simulated model (red) to the measured data (blue)

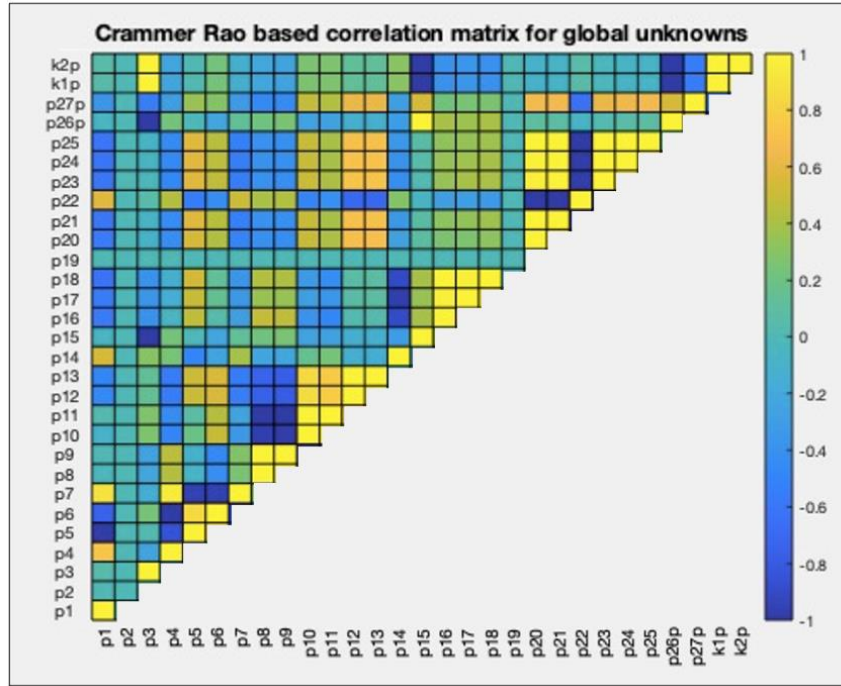
The results obtained show an adequate fit for both HdmX and Hdm2 proteins where the simulated data lies within the 20% error criterion at each 60 minute data point, yet this was not for P53 at 180 and 360 minutes as well as for hdm2 mRNA at 60 and 360 minutes. More

importantly, this parameter estimation optimization problem was not solved as convergence was not achieved after over 5000 iterations which translates to upwards of 55 hours of computation time. The parameter values for this model are detailed in APPENDIX A.

Consequently, we reduced the model to 5 state variables and tested a log likelihood approach. By minimizing the model, we removed Wip1 mRNA and simplified the term for Wip1 protein production that was dependent on this 6th state variable. After receiving promising results from Simulink, yet lacking the ability to extract information such as the covariance matrix, we utilized the MATLAB package AMIGO and programmed the 5-state variable model, selecting the log likelihood cost function to minimize and a metaheuristic of local and global optimization algorithms for this minimization to obtain the following results.



Parameter estimation results of 5-state variable Wip1 protein model with 20% confidence interval



Cramer Rao based correlation matrix for 5 state variable Wip1 model

This approach results in a significantly better fit where the simulated curves lie within a 20% confidence interval for all points except for Hdm2 mRNA at 360 minutes. To determine the structural identifiability of the model, we analyzed the covariance matrix that displays the correlation between parameters where a yellow or dark blue block signals high correlation and poor numerical identifiability, also referenced quantitatively in APPENDIX B. In this model, there are 8 sets of correlated parameters: p4-p6, p18-p11, p10-p13, p14-p16, p16-p18, and p23-p25, p3 and k1-k2, and p15-p26. This high degree of correlation suggests the model is overparametrized and should be reduced by fixing parameters to constant values or eliminating them. We identified these parameters as either multiple magnitudes of order smaller than other parameters within the same term or if they rank in the bottom 10% of the parameters that affect the outcome of the optimization based on a sensitivity analysis. We reduced the 5-state variable model to the to a 23 parameter model compared to 31 parameters by eliminating p03, p09, p23 and fixing p05 and p17.

$$\frac{dx_1}{dt} = p_1 x_4 (t - \tau_1) - p_4' x_1^2 \left(1 + \frac{p_6 u}{p_7 + u}\right) + k_{15}' x_5 x_1$$

where $p_4' = \frac{p_4}{p_5}$ and $k_{15}' = k_{15} x_5(\max)$

$$\frac{dx_2}{dt} = p_8 - p_{10} \left(\frac{x_1 x_2}{p_{11} + x_2} \left(1 + \frac{p_{12} u}{p_{13} + u}\right) \right)$$

$$\frac{dx_3}{dt} = p_{14} - p_{15} x_3 - p_{16}' x_1 (1 - p_{18} u) - k_{35}' x_5 x_3$$

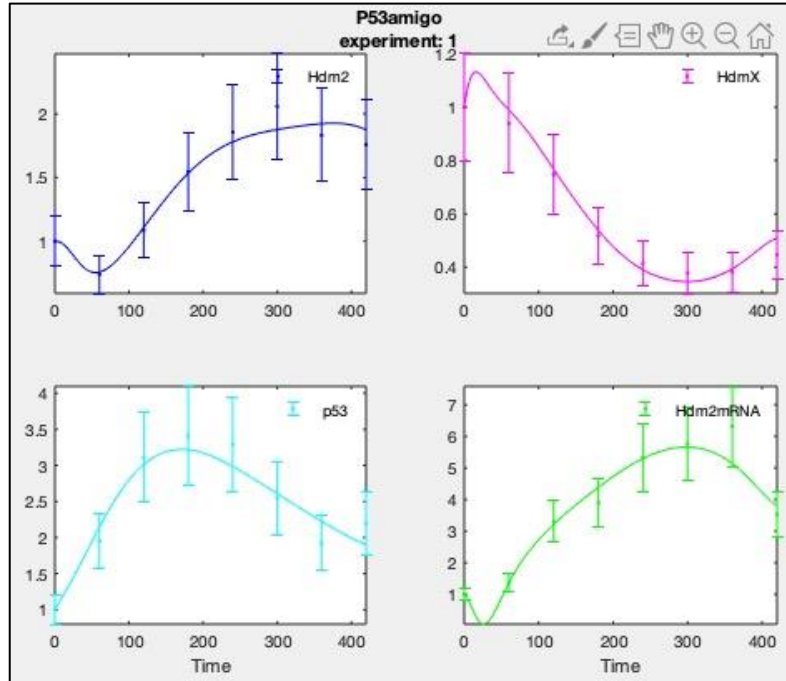
where $p_{16}' = \frac{p_{16}}{p_{17}}$ and $k_{35}' = k_{35} x_5(\max)$

$$\frac{dx_4}{dt} = p_{19} - p_{20} x_4 + \frac{p_{21} x_3^4}{p_{22}^4 + x_3^4} (1 - p_{24} x_1) (1 - p_{25} x_2)$$

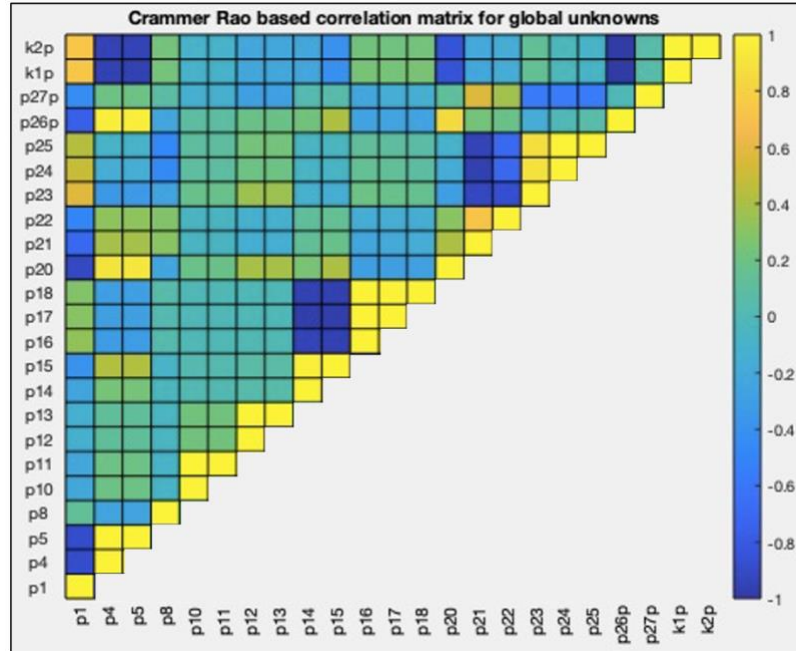
$$\frac{dx_5'}{dt} = p_{26}' - p_{27} x_5'$$

where $p_{26}' = \frac{p_{26}}{x_5(\max)}$ and $x_5' = \frac{x_5}{x_5(\max)}$

After performing parameter estimation through both Simulink and AMIGO2, the results in AMIGO2 with a log likelihood cost function and a combination of local and global optimization algorithms resulted in the better fit.



Parameter estimation results of reduced 5-state variable Wip1 protein model with 20% confidence interval



Cramer Rao based correlation matrix for reduced 5-state variable Wip1 protein model

Although the correlation matrix displays fewer sets of parameter combinations, the fit for the observables appears to be less accurate indicating a poorer fit. This suggests that the reduction of certain parameters while maintaining 5 state variables causes the model to lose power in its ability to capture the dynamics of the p53 regulation network for normally proliferating WS1 cells.

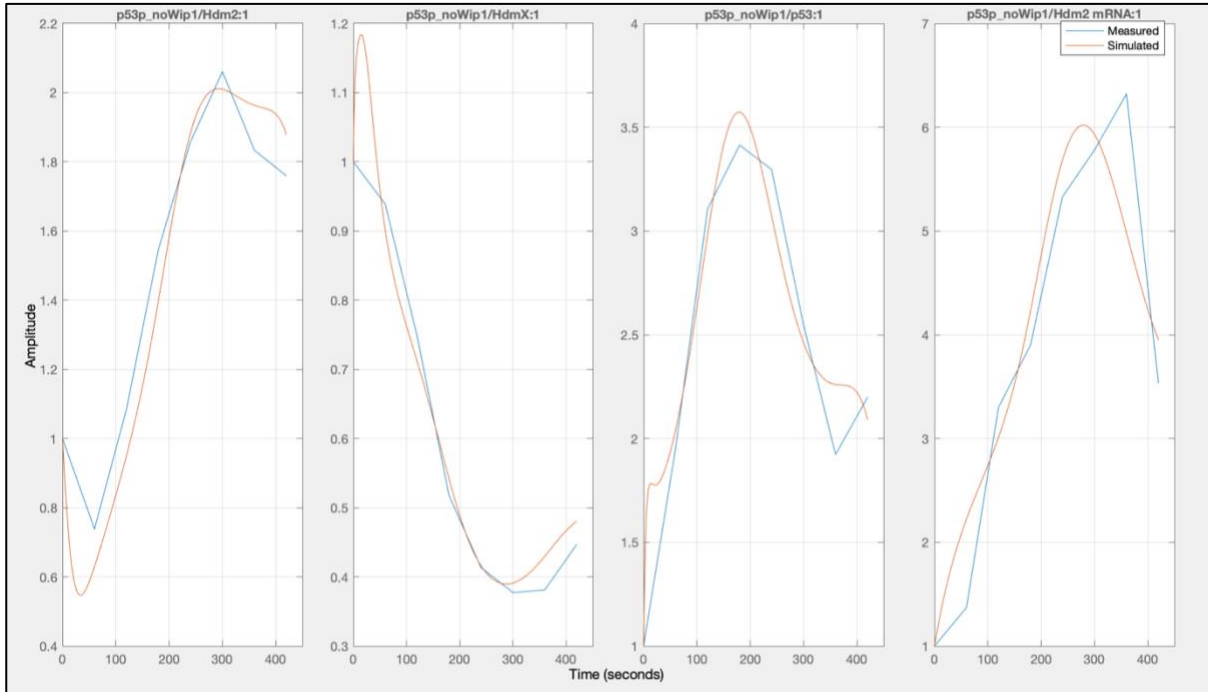
Best fit for P53 expansion

We also explored the expansion of the P53 protein as an alternative to the negative regulator Wip1. p53 is found in 4 biological states in the cell: unphosphorylated, singly phosphorylated, doubly phosphorylated, and acetylated. We expanded the total P53 state variable into its unphosphorylated form x_3 and phosphorylated x_5 where x_5 represents the combination or the single and double phosphorylated state.

$$\frac{dx_3}{dt} = p_{14} - p_{15}x_3 - p_{16}x_1x_3 - p_{33}x_2x_3 + p_{30}x_6 - p_{34}ux_3 \quad (10)$$

$$\frac{dx_5}{dt} = p_{34}x_3u - p_{30}x_6 \quad (12)$$

We performed parameter estimation in Simulink to estimate the values of 18 parameters using the sum of squared errors as the cost function which was minimized by the Simplex algorithm. The optimization problem converged based on a stringent function tolerance of $1e^{-5}$ after 1736 iterations and 7 hours of computation time.



Parameter estimation results of 5-state variable model with expanded P53 protein fitting the simulated model (red) to the measured data (blue)

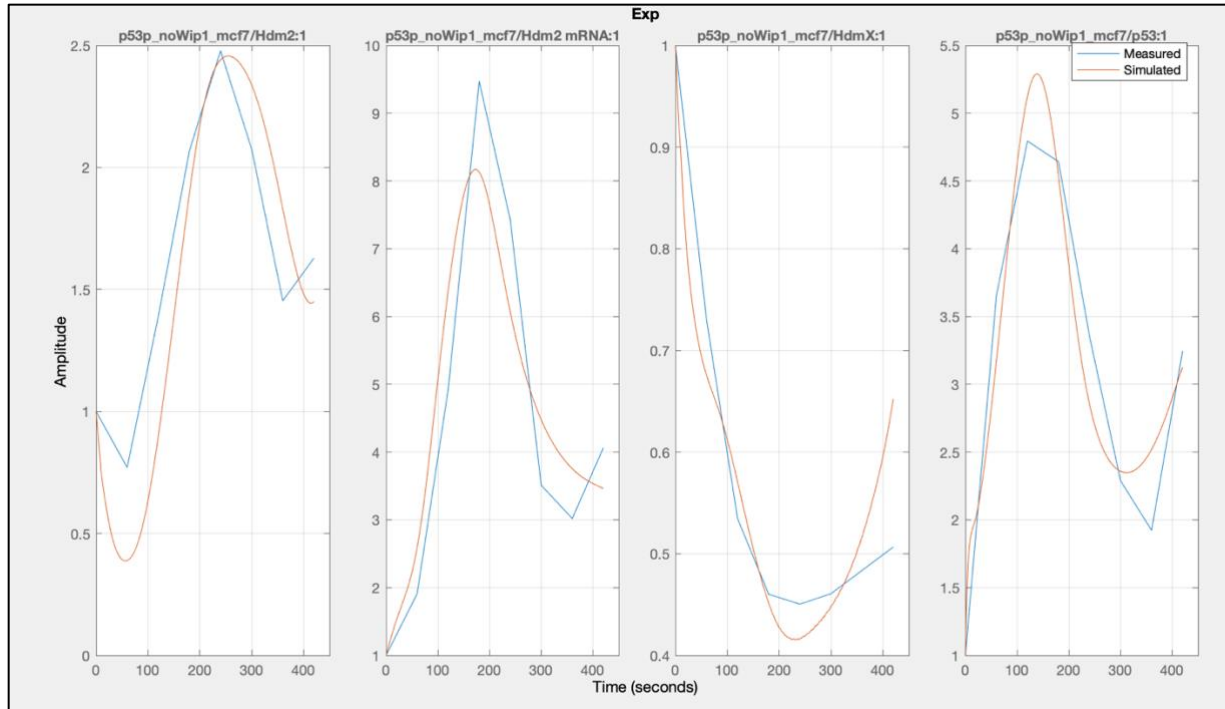
p01	Hdm2 basal production rate	0.0237	time ⁻¹
p03	Hdm2 basal degradation rate	7.6203e-4	time ⁻¹
p04	Hdm2 phosphorylation by ATM	0.0514	time ⁻¹
p05	Hdm2-HdmX complex association rate	0.0496	concentration
p08	HdmX basal production rate	0.5877	unitless
p09	HdmX basal degradation rate	0.0393	time ⁻¹
p10	HdmX phosphorylation by ATM	0.5122	time ⁻¹
p11	Hdm2-HdmX complex association rate	0.4738	concentration
p14	P53 basal production rate	0.0301	time ⁻¹
p15	P53 basal degradation rate	0.0130	time ⁻¹
p16	P53 degradation rate (Hdm2 dependent)	0.0108	time ⁻¹
p19	hdm2 mRNA production rate	0.0463	time ⁻¹
p20	hdm2 mRNA basal degradation rate	0.0157	time ⁻¹
p21	hdm2 mRNA transcription rate (p53 dependent)	0.1798	time ⁻¹
p22	Hill K of hdm2 mRNA transcription (p53 dependent)	5.7586	unitless
p30	P53p basal degradation rate + P53p dephosphorylation by phosphatase	0.2136	concentration
p33	P53 degradation rate (HdmX dependent)	0.0274	time ⁻¹
p34	P53 phosphorylation by ATM	0.358	conc time ⁻¹

Parameter estimation values of 5-state variable model with expanded P53 protein

This model achieved a simulated fit that lies within the 20% confidence interval for every measured data point of Hdm2, HdmX, P53 and hdm2 mRNA. We also performed parameter estimation using AMIGO2 to obtain the correlation matrix and determine the identifiability of the model yet the curve-fitting displayed a reduced accuracy as it did not remain within the 20% confidence interval for 2 data points and the correlation matrix provided in APPENDIX B displays a high correlation between 4 sets of parameters in the Hdm2 and P53 state variables.

Given the favorable fit of this candidate model in Simulink, we performed parameter estimation on data from the cancerous cell line MCF7 to examine the robustness of the model to capture the biology of the P53 network in another biological context. The optimization problem converged after 2067 iterations using a cost function of sum squared error solved with the Simplex

algorithm yet only converged with a minimal function tolerance of $1e^{-3}$ and displaying an adequate fit with peaks in all 4 state variables either overshoot or undershot.



Parameter estimation results of 5-state variable model with expanded P53 protein fitting the simulated model (red) to the cancerous MCF7 measured data (blue)

Discussion

We explored 3 different dynamical models for the P53 regulatory network with the negative regulatory Wip1 and the expansion of the total P53 protein into its unphosphorylated and phosphorylated forms. We analyzed the candidate parameter sets using a 20% coefficient of variation for best fit between simulated curves and measured data and correlation matrices to determine the numerical identifiability of proposed model. We also reduced the candidate models to design the simplest model to avoid overparameterization while capturing the biology of the network using a sensitivity analysis to reduce correlation of parameters in a complex model. These multiple methods provide a systematic approach to improve the quality of the candidate model by reducing the uncertainty of the parameter estimation solution.

Utilizing the criteria of best fit and parameter correlation matrices, the P53 expansion provided a better fit for the WS1 normally proliferating data than the incorporation of the Wip1 protein and Wip1 mRNA. The model we started with consisted of 31 parameters and 6 state variables and could not be solved as shown by a lack of convergence after over 40 hours of computation for a minimal function tolerance of 0.001. This severely reduces the reliability of this model as the parameter set produced is not guaranteed to be reproducible and suggests the model is overparametrized and should be reduced. Incorporating the wip1 mRNA state variable as a standalone production term in the Wip1 state variable ODE leading to a 5-state variable model achieved convergence and a best fit with only 4 data points out of 32 remaining outside of the 20% confidence interval. In comparison, the expansion of P53 into its unphosphorylated and phosphorylated form achieved convergence in a smaller number of iterations where every measured WS1 datapoint showed a smaller residual at the measured time intervals and remained within the 20% confidence interval. Given this analysis, the expansion of P53 leads to a better fit

for normally proliferating WS1 skin model and should be the model further explored in the context of cancerous MCF7 cells.

One drawback from these current models is the lack of predictive power due to the small amount of biological data points provided. We increased the number of points by combining simulated data points between the 60 minute measured intervals and attempted to perform parameter estimation on only the first half of the data. We aligned the resulting curves with the second half of the data to determine the predictive power of the optimization results, but the curves lied heavily outside of the 20% confidence interval for the second half of the data points, indicating a need for more biological data to power prediction. In addition, the limited dataset does not allow for validation of the model on another set of datapoints which would provide feedback on the robustness of the candidate model. Therefore, no conclusions can be drawn currently as the best fit for this model was not achieved with the methods explored in this investigation.

In this project we explored two alternative proteins, Wip1 and phosphorylated P53, as key proteins in the P53 regulatory network with the goal of identifying “druggable targets” within the network. Given the importance of this particular gene, mutated in over 50% of all cancers, it is valuable to explore the addition of other proteins including ATM kinase, which we included through input data to this particular model as well as further expanding the complexity of P53 into its doubly phosphorylated form, two proteins that have been extensively reported on in the literature for their role in this network. We also propose investing the validation of this model against another data set that can be generated through methods such as Monte Carlo simulations to further develop the robustness of this model in the goal of increasing reproducibility and reliability.

Files used to generate all results provided can be supplied by emailing alazrak1@cs.ucla.edu.

APPENDIX A

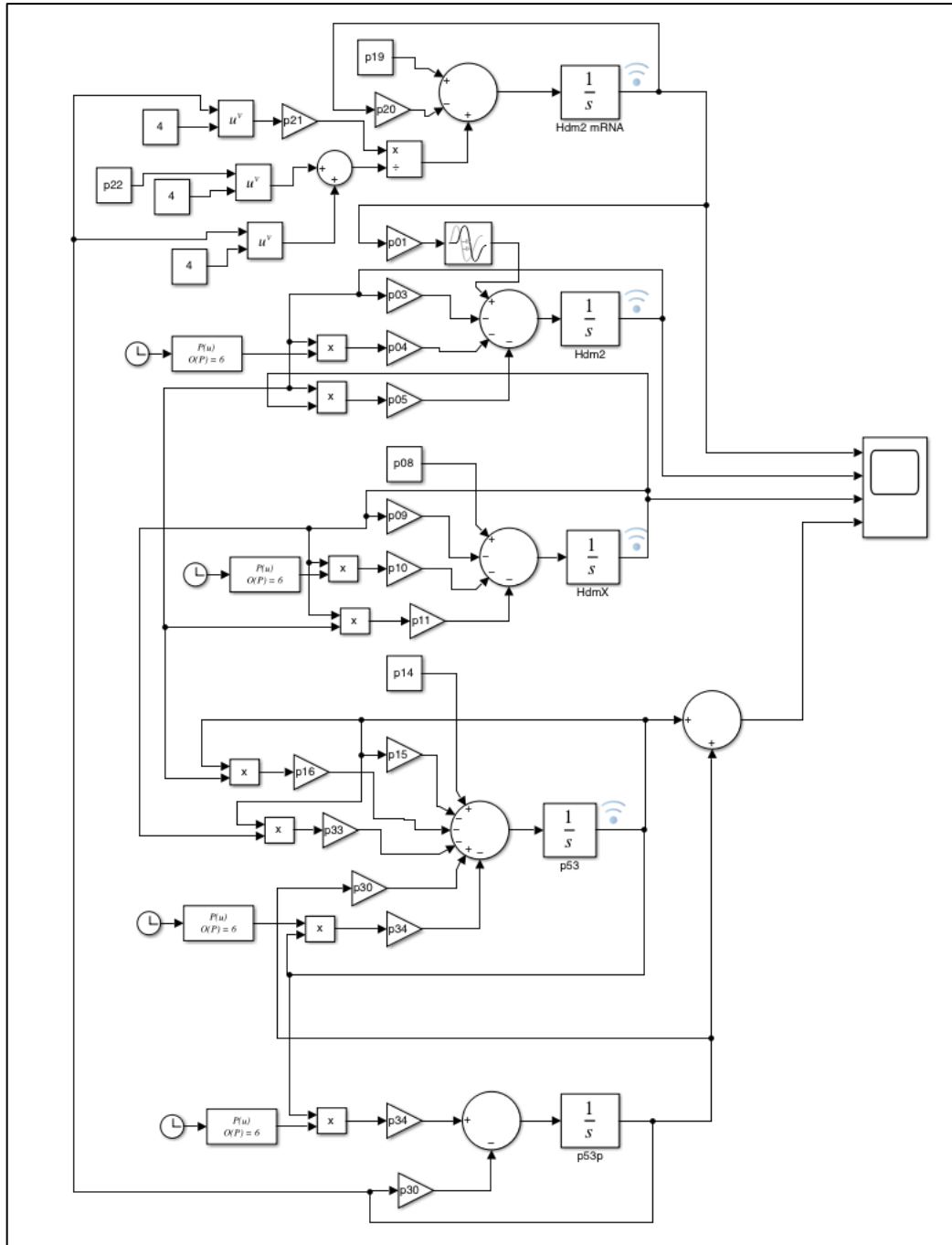
Estimated parameter values for the 6 state variable model with Wip1 protein and Wip1 mRNA

p01	Hdm2 production rate	0.0093	time ⁻¹
p02	Hdm2 transcription and transport delay	0	time ⁻¹
p03	Hdm2 basal degradation rate	0.0030	time ⁻¹
p04	Hdm2 degradation rate by ubiquitination	0.0757	time ⁻¹
p05	Hill K for Hdm2 ubiquitination	107.819	concentration
p06	Hill V for Hdm2 ubiquitination	81.5774	unitless
p07	Hill K for Hdm2 phosphorylation by kinase	1.71	concentration
p08	HdmX production rate	0.0185	time ⁻¹
p09	HdmX basal degradation rate	0.0091	time ⁻¹
p10	HdmXdegradation rate (Hdm2 dependent)	0.0231	time ⁻¹
p11	Hill K for HdmX ubiquitination	2.2174	concentration
p12	Hill V for HdmX ubiquitination	3.6902	unitless
p13	Hill K for HdmX phosphorylation by kinase	1.0343	concentration
p14	p53 production rate	0.0979	time ⁻¹
p15	p53 basal degradation rate	0.0076	time ⁻¹
p16	p53 degradation rate (Hdm2 dependent)	0.0356	time ⁻¹
p17	Hill K for p53 kinase phosphorylation	3.7754E-06	concentration
p18	Inhibition of p53 degradation by kinase phosphorylation	0.4026	unitless
p19	hdm2 mRNA production rate	0.0886	time ⁻¹
p20	hdm2 mRNA basal degradation rate	0.0412	time ⁻¹
p21	hdm2 mRNA transcription rate (p53 dependent)	1.6949	time ⁻¹
p22	Hill K of hdm2 mRNA transcription (p53 dependent)	3.76	unitless
p23	Stimulation of mRNA transcription by p53 phosphorylation	3.5899E-06	unitless
p24	Inhibition of mRNA transcription by Hdm2	0.1204	unitless
p25	Inhibition of mRNA transcription by HdmX	1.1916	unitless
p26	Wip1 translation rate	0.0199	time ⁻¹
p27	Wip1 basal degradation rate	0.0573	time ⁻¹
p28	Wip1 mRNA basal production rate	0.0019	conc time ⁻¹
p29	Wip1 mRNA transcription rate (P53 dependent)	0.00062481	conc time ⁻¹
p30	Wip1 mRNA transcription Michaelis constant	0.1840	concentration
p31	Wip1 mRNA degradation rate	0.0562	time ⁻¹
k15	Wip1 induced Mdm2 dephosphorylation	0.538	conc time ⁻¹
k35	Wip1 induced P53 dephosphorylation	0.9469	conc time ⁻¹
τ_1	Wip1 and Hdm2 mRNA translation delay	10	time
τ_2	Wip1 transcription delay	30	time

p16	0.070	0.722	0.613	-0.021	0.830	0.111	-0.191	0.060	0.481	0.083	1.000	
p15	0.478	0.601	0.355	-0.544	0.810	0.065	0.818	0.059	0.058	1.000		
p14	0.401	0.733	0.181	0.267	-0.144	0.809	0.700	0.359	1.000			
p11	0.709	-0.198	0.143	0.134	-0.189	0.152	0.943	1.000				
p10	-0.589	0.359	0.102	0.115	0.153	0.318	1.000					
p09	0.087	0.712	0.643	0.785	0.078	1.000						
p08	0.791	-0.644	0.093	0.398	1.000							
p05	0.644	0.918	0.921	1.000								
p04	-0.223	0.877	1.000									
p03	0.687	1.000										
p01	1.000											

APPENDIX C

Example Simulink block diagram



Example code for 6-state variable model in AMIGO2

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%
% TITLE: p53 Parameter estimation
```

```

%
% Adapted from the NFkB example provided
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%=====
% PATHS RELATED DATA
%=====
inputs.pathd.results_folder='P53amigo'; % Name of the folder to
keep results
inputs.pathd.short_name='P53amigo'; % To be used for problem
results folders and files

%=====
% MODEL RELATED DATA
%=====
inputs.model.input_model_type='charmodelC'; % Model introduction:
'charmodelC'|'c_model'|'charmodelM'|'matlabmodel'|'sbmlmodel'|
%
'blackboxmodel'|'blackboxcost
inputs.model.n_st=6; % Number of states
inputs.model.n_par=33; %EBC; +2 includes tau1 and tau2 % Number of parameters
in the model
inputs.model.n_stimulus=0; % Number of stimulus in the
model
inputs.model.st_names=char('p53','Hdm2','HdmX','Hdm2mRNA','Wip1','Wip1mRNA');
% inputs.model.par_names=char('p1','p2','p3','p4','p5','p6','p7','p8','p9','p10',...
%
'p11','p12','p13','p14','p15','p16','p17','p18','p19','p20','p21',...
%
'p22','p23','p24','p25','p26','p27','p28','p29','p30','p31',...
%
'k1','k2','tau1','tau2'); %EBC tau1 and tau2 added
inputs.model.par_names=char('p1','p2','p3','p4','p5','p6','p7','p8','p9','p10',...
'p11','p12','p13','p14','p15','p16','p17','p18','p19','p20','p21',...
'p22','p23','p24','p25','p26p','p27p','p28','p29','p30','p31p',...
'k1p','k2p');

% Names of the stimulus
inputs.model.stimulus_names=[]; %Serine 15 phosphorylation as input
% Eqns (equations): ODEs or
DAEs % ODEs:
% ydot =f (ydot,y,u,theta,t)
% DAES:
% delta=ydot-f(ydot,y,u,

inputs.model.eqns=char('u = (5.1989e-16.*(t.^6))+(3.0777e-12.*(t.^5))-(3.88785e-
09.*(t.^4))+(1.60323e-06.*(t.^3))-(2.81557e-04.*(t.^2))+(0.022275.*(t))+0.0049857',...
%
'dHdm2=p1*Hdm2mRNA*(t-tau1)-p3*Hdm2-
p4*((Hdm2^2)/(p5+Hdm2))*(1+((p6*u)/(p7+u)))+(k1*Wip1*Hdm2)',...
%
'dHdmX=p8-(p9*HdmX)-
p10*((Hdm2*HdmX)/(p11+HdmX))*(1+((p12*u)/(p13+u)))',...
%
'dp53=p14-(p15*p53)-p16*((Hdm2*p53)/(p17+p53))*(1-(p18*u))-
(k2*Wip1*p53)',...
%
'dHdm2mRNA=p20+p21*((1-p24)*(1-p25))/(p22^4+1))-
p20*Hdm2mRNA+((p21*(p53^4)/(p22^4+(p53^4)))*(1+p23*u)*(1-p24*Hdm2)*(1-p25*HdmX)',...
%
'dWip1=p26*Wip1mRNA*(t-tau1)-(p27*Wip1)',...
%
'dWip1mRNA=p28+p29*((p53*(t-tau2)^4)/(p30^4+p53*(t-
tau2)^4))-(p31*Wip1mRNA)');

```

```

% inputs.model.eqns=char('u = (5.1989e-16*(t^6))+(3.0777e-12*(t^5))-(3.88785e-
09*(t^4))+(1.60323e-06*(t^3))-(2.81557e-04*(t^2))+(0.022275*(t))+0.0049857',...
% 'dHdm2=p1*Hdm2mRNA-p3*Hdm2-
p4*((Hdm2^2)/(p5+Hdm2))*(1+((p6*u)/(p7+u)))+(k1*Wip1*Hdm2)',...%tau1 was removed
% 'dHdmX=p8-(p9*HdmX)-
p10*((Hdm2*HdmX)/(p11+HdmX))*(1+((p12*u)/(p13+u)))',...
% 'dp53=p14-(p15*p53)-p16*((Hdm2*p53)/(p17+p53))*(1-(p18*u))-
(k2*Wip1*p53)',...
% 'dHdm2mRNA=p20-p21*((1-p24)*(1-p25))/(p22^4+1))-
p20*Hdm2mRNA+((p21*(p53^4))/(p22^4+(p53^4)))*(1+p23*u)*(1-p24*Hdm2)*(1-p25*HdmX)',...
% 'dWip1=p26*Wip1mRNA-(p27*Wip1)',... %tau1 was removed
% 'dWip1mRNA=p28+p29*((p53^4)/((p30^4)+(p53^4)))-
(p31*Wip1mRNA)'); %tau2 was removed
inputs.model.eqns=char('u = (5.1989e-16*(t^6))+(3.0777e-12*(t^5))-(3.88785e-
09*(t^4))+(1.60323e-06*(t^3))-(2.81557e-04*(t^2))+(0.022275*(t))+0.0049857',...
% 'dHdm2=p1*Hdm2mRNA-p3*Hdm2-
p4*((Hdm2^2)/(p5+Hdm2))*(1+((p6*u)/(p7+u)))+(k1p*Wip1*Hdm2)',...%tau1 was removed
% 'dHdmX=p8-(p9*HdmX)-
p10*((Hdm2*HdmX)/(p11+HdmX))*(1+((p12*u)/(p13+u)))',...
% 'dp53=p14-(p15*p53)-p16*((Hdm2*p53)/(p17+p53))*(1-(p18*u))-
(k2p*Wip1*p53)',...
% 'dHdm2mRNA=p20-p21*((1-p24)*(1-p25))/(p22^4+1))-
p20*Hdm2mRNA+((p21*(p53^4))/(p22^4+(p53^4)))*(1+p23*u)*(1-p24*Hdm2)*(1-p25*HdmX)',...
% 'dWip1=p26p*Wip1mRNA-(p27p*Wip1)',... %tau1 was removed
% 'dWip1mRNA=p28+p29*((p53^4)/((p30^4)+(p53^4)))-
(p31p*Wip1mRNA)'); %tau2 was removed
%
inputs.model.par=[0.0093,0,0.0030,0.0757,107.8193,81.5774,1.71,0.0185,0.0091,0.0231,..
.
% 2.2174,3.6902,1.0343,0.0979,0.0076,0.0356,3.7754e-06,0.4026,0.0886,...
% 0.0412,1.6949,3.76,3.5899e-06,0.1204,1.1916,0.0199,0.0573,0.0019,...
% 0.00062481,0.1840,0.0562,0.0527,0.9476,30,10];
% Nominal parameter values when available
% EBC two extra values added for tau1, tau2
inputs.model.par=[0.0093,0,0.0030,0.0757,107.8193,81.5774,1.71,0.0185,0.0091,0.0231,..
.
2.2174,3.6902,1.0343,0.0979,0.0076,0.0356,3.7754e-06,0.4026,0.0886,...
0.0412,1.6949,3.76,3.5899e-06,0.1204,1.1916,0.0199*0.04,0,0.0019,...
0.00062481,0.1840,0.0562*0.04,0,0];

%=====
% EXPERIMENTAL SCHEME RELATED DATA
%=====

inputs.exps.n_exp=1;

% EXPERIMENT 1
inputs.exps.n_obs{1}=4;
inputs.exps.obs_names{1}=char('Hdm2','HdmX','p53','Hdm2mRNA');% names of observables
inputs.exps.obs{1}=char('Hdm2','HdmX','p53','Hdm2mRNA');
inputs.exps.exp_y0{1}= [1 1 1 1 1 1];

inputs.exps.t_f{1}=420;%in seconds

inputs.exps.u_interp{1}=[]; %Stimuli definition for experiment 1:
%OPTIONS:u_interp: 'sustained'
|'step'|'linear'(default)|'pulse-up'|'pulse-down'
inputs.exps.t_con{1}=[]; % Input swithching times: Initial and
final time
inputs.exps.u{1}=[];
inputs.exps.n_s{1}=8; % [] Number of sampling times for each
experiment.

```

```

%=====
% EXPERIMENTAL DATA RELATED INFO
%=====

%Experimental data 1:
%this is the matrix of the 4 outputs (Hdm2, HdmX, p53, Hdm2 mRNA)
inputs.exps.exp_data{1}=[
    1          1          1          1
    0.738453592 0.938706211 1.953239036 1.37
    1.084105847 0.747776102 3.106851548 3.3
    1.546930484 0.517499932 3.413333746 3.9
    1.857195829 0.413398929 3.29626695 5.33
    2.060705374 0.37729276 2.54531142 5.78
    1.833295011 0.381146764 1.922284468 6.32
    1.759141834 0.446707728 2.199259236 3.53
];

%
% Error data 1:
% Standard deviation: 20% for all
inputs.exps.error_data{1}=0.2.*[
    1          1          1          1
    0.738453592 0.938706211 1.953239036 1.37
    1.084105847 0.747776102 3.106851548 3.3
    1.546930484 0.517499932 3.413333746 3.9
    1.857195829 0.413398929 3.29626695 5.33
    2.060705374 0.37729276 2.54531142 5.78
    1.833295011 0.381146764 1.922284468 6.32
    1.759141834 0.446707728 2.199259236 3.53
];

%=====
% UNKNOWNNS RELATED DATA
%=====

% GLOBAL UNKNOWNNS (SAME VALUE FOR ALL EXPERIMENTS)

%we only have one experiment so I don't think this is relevant

inputs.PEsol.id_global_theta=char('p1','p3','p4','p5','p6','p7','p8','p9','p10',...
    'p11','p12','p13','p14','p15','p16','p17','p18','p20','p21',...
    'p22','p23','p24','p25','p26p','p28','p29','p30','p31p');

% 'all'|User selected
inputs.PEsol.global_theta_max=10.*[0.0093,0.0030,0.0757,107.8193,81.5774,1.71,0.0185,0
.0091,0.0231,...
    2.2174,3.6902,1.0343,0.0979,0.0076,0.0356,3.7754e-06,0.4026,... %0.0886,...
    0.0412,1.6949,3.76,3.5899e-06,0.1204,1.1916,0.0199*0.04,0.0019,...
    0.00062481,0.1840,0.0562*0.04];
% Maximum allowed values for the paramters: 10 times parameter space
inputs.PEsol.global_theta_min=1e-10.*ones(1,28); %Minimum allowed values : non
negative

inputs.PEsol.init_steady_state='off';
%=====
% COST FUNCTION RELATED DATA
%=====

inputs.PEsol.PEcost_type='lsq'; % 'lsq' (weighted least squares
default) | 'llk' (log likelihood) | 'user_PEcost'
inputs.PEsol.lsqr_type='Q_mat'; % [] To be defined for llk
function, 'homo' | 'homo_var' | 'hetero'
inputs.PEsol.lsqr_Qmat{1}= 0.04.*[

```

```

1          1          1          1
0.738453592 0.938706211 1.953239036 1.37
1.084105847 0.747776102 3.106851548 3.3
1.546930484 0.517499932 3.413333746 3.9
1.857195829 0.413398929 3.29626695 5.33
2.060705374 0.37729276 2.54531142 5.78
1.833295011 0.381146764 1.922284468 6.32
1.759141834 0.446707728 2.199259236 3.53
].^2; % the weighting matrix (an 8x4) for {iex}

%=====
% NUMERICAL METHODS RELATED DATA
%=====
%
% DEFAULTS ARE BEING USED FOR SIMULATION
inputs.nlpsol.nlpsolver='ess'; %add in solver here % [] NLP solver: %n2fb
levenberg marquardt, dhc = direct hill climbing, no gradient, nl2sol and lsqnonlin is
levenberg marquardt
%
% LOCAL:
'local_fmincon'|'local_n2fb'|'local_dn2fb'|'local_dhc'|
%
'local_ipopt'|'local_solnp'|'local_nomad'|'local_nl2sol'|
%
% 'local_lsqnonlin'
%
MULTISTART:'multi_fmincon'|'multi_n2fb'|'multi_dn2fb'|'multi_dhc'|
%
'multi_ipopt'|'multi_solnp'|'multi_nomad'|'multi_nl2sol'|
%
% 'multi_lsqnonlin'
%
% GLOBAL: 'de' | 'sres'
%
% HYBRID:
'hyb_de_fmincon'|'hyb_de_n2fb'|'hyb_de_dn2fb'|'hyb_de_dhc'|'hyb_de_ipopt'|
%
'hyb_de_solnp'|'hyb_de_nomad'|
%
'hyb_sres_fmincon'|'hyb_sres_n2fb'|'hyb_sres_dn2fb'|'hyb_sres_dhc'|
%
'hyp_sres_ipopt'|'hyb_sres_solnp'|'hyb_sres_nomad'
%
% METAHEURISTICS:
% 'ess' or 'eSS' (default)
% Note that the corresponding
defaults are in files:
%
% OPT_solvers\DE\de_options.m;
OPT_solvers\SRES\sres_options.m;
%
OPT_solvers\eSS_**\ess_options.m
%

inputs.nlpsol.eSS.log_var = 1:length(inputs.PEsol.global_theta_guess); % Decision
variables to be considered in log-scale. Suitable for this problem with parameters
% ranging from
0 to 1.
inputs.nlpsol.eSS.maxeval = 300000; %maximum number of evaluations
inputs.nlpsol.eSS.maxtime = 300; %max computational effort
inputs.nlpsol.eSS.log_var = [1:1:length(inputs.PEsol.global_theta_max)];
inputs.nlpsol.eSS.local.solver = 'lsqnonlin'; % 'nl2sol' uses gradient, lsqnonlin,
check that always converging to same solution
inputs.nlpsol.eSS.local.finish = 'fminsearch';

Code for modeling 5 state variable model with log likelihood approach:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% TITLE: p53 Parameter estimation
%
```

```

% Adapted from the NFkB example provided
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%=====
% PATHS RELATED DATA
%=====
inputs.pathd.results_folder='P53amigo'; % Name of the folder to
keep results
inputs.pathd.short_name='P53amigo'; % To be used for problem
results folders and files

%=====
% MODEL RELATED DATA
%=====
inputs.model.input_model_type='charmodelC'; % Model introduction:
'charmodelC'|'c_model'|'charmodelM'|'matlabmodel'|'sbmlmodel'|
%
'blackboxmodel'|'blackboxcost
inputs.model.n_st=5; % Number of states
inputs.model.n_par=29; %EBC; +2 includes tau1 and tau2 % Number of parameters
in the model
inputs.model.n_stimulus=0; % Number of stimulus in the
model
inputs.model.st_names=char('p53','Hdm2','HdmX','Hdm2mRNA','Wip1');
inputs.model.par_names=char('p1','p2','p3','p4','p5','p6','p7','p8','p9','p10',...
'p11','p12','p13','p14','p15','p16','p17','p18','p19','p20','p21',...
'p22','p23','p24','p25','p26p','p27p','k1p','k2p');
inputs.model.stimulus_names=[];

inputs.model.eqns=char('u = (5.1989e-16*(t^6))+(3.0777e-12*(t^5))-(3.88785e-
09*(t^4))+(1.60323e-06*(t^3))-(2.81557e-04*(t^2))+(0.022275*(t))+0.0049857',...
'dHdm2=p1*Hdm2mRNA-p3*Hdm2-
p4*((Hdm2^2)/(p5+Hdm2))*(1+((p6*u)/(p7+u)))+(k1p*Wip1*Hdm2)',...%tau1 was removed
'dHdmX=p8-(p9*HdmX)-
p10*((Hdm2*HdmX)/(p11+HdmX))*(1+((p12*u)/(p13+u)))',...
'dp53=p14-(p15*p53)-p16*((Hdm2*p53)/(p17+p53))*(1-(p18*u))-
(k2p*Wip1*p53)',...
'dHdm2mRNA=p20-p21*((1-p24)*(1-p25))/(p22^4+1))-
p20*Hdm2mRNA+((p21*(p53^4))/(p22^4+(p53^4)))*(1+p23*u)*(1-p24*Hdm2)*(1-p25*HdmX)',...
'dWip1=p26p-(p27p*Wip1)'); %tau2 was removed

inputs.model.par=[0.0093,0,0.0030,0.0757,107.8193,81.5774,1.71,0.0185,0.0091,0.0231,...
2.2174,3.6902,1.0343,0.0979,0.0076,0.0356,3.7754e-06,0.4026,0.0886,0.0412,...
1.6949,3.76,3.5899e-06,0.1204,1.1916,0.0199*0.04,0,0,0];

inputs.exps.t_f{1}=420; %in seconds

inputs.exps.u_interp{1}=[];%Stimuli definition for experiment 1:
%OPTIONS:u_interp: 'sustained'
|'step'|'linear'(default)|'pulse-up'|'pulse-down'
inputs.exps.t_con{1}=[]; % Input swithching times: Initial and final time
inputs.exps.u{1}=[];
inputs.exps.n_s{1}=8; % [] Number of sampling times for each experiment.

%=====
% EXPERIMENTAL SCHEME RELATED DATA
%=====

```

```

inputs.exps.n_exp=1;

% EXPERIMENT 1
inputs.exps.n_obs{1}=4;
inputs.exps.obs_names{1}=char('Hdm2','HdmX','p53','Hdm2mRNA');% names of observables
inputs.exps.obs{1}=char('Hdm2','HdmX','p53','Hdm2mRNA');
inputs.exps.exp_y0{1}= [1 1 1 1 1];

%=====
% EXPERIMENTAL DATA RELATED INFO
%=====

%Experimental data 1:
%this is the matrix of the 4 outputs (Hdm2, HdmX, p53, Hdm2 mRNA)
inputs.exps.exp_data{1}=[
    1          1          1          1
    0.738453592 0.938706211 1.953239036 1.37
    1.084105847 0.747776102 3.106851548 3.3
    1.546930484 0.517499932 3.413333746 3.9
    1.857195829 0.413398929 3.29626695 5.33
    2.060705374 0.37729276 2.54531142 5.78
    1.833295011 0.381146764 1.922284468 6.32
    1.759141834 0.446707728 2.199259236 3.53

];

%
% Error data 1:
% Standard deviation: 20% for all
inputs.exps.error_data{1}=0.2.*[
    1          1          1          1
    0.738453592 0.938706211 1.953239036 1.37
    1.084105847 0.747776102 3.106851548 3.3
    1.546930484 0.517499932 3.413333746 3.9
    1.857195829 0.413398929 3.29626695 5.33
    2.060705374 0.37729276 2.54531142 5.78
    1.833295011 0.381146764 1.922284468 6.32
    1.759141834 0.446707728 2.199259236 3.53

];

%=====
% UNKNOWNNS RELATED DATA
%=====
% GLOBAL UNKNOWNNS (SAME VALUE FOR ALL EXPERIMENTS)

%we only have one experiment so I don't think this is relevant

inputs.PEsol.id_global_theta=char('all');

% Minimum allowed values for the parameters
inputs.PEsol.global_theta_min=[0 0 0 0 0 0 0 0 0 0 ... %1-10
                                0 0 0 0 0 0 0 0 0 0 ... %11-20
                                0 0 0 0 0 0 0 0 0 0 ... %21-29
                                ];

inputs.PEsol.global_theta_max=100.*[0.0093,1e-
10,0.0030,0.0757,107.8193,81.5774,1.71,0.0185,0.0091,0.0231,...
2.2174,3.6902,1.0343,0.0979,0.0076,0.0356,3.7754e-06,0.4026,0.0886,...
0.0412,1.6949,3.76,3.5899e-06,0.1204,1.1916,0.0199*0.04,1e-10,1e-10,1e-10];

inputs.PEsol.init_steady_state='off';

%=====
% COST FUNCTION RELATED DATA

```

```

%=====

inputs.PEsol.PEcost_type='llk';
inputs.PEsol.llk_type='homo_var';

%=====
% NUMERICAL METHODS RELATED DATA
%=====
%
% DEFAULTS ARE BEING USED FOR SIMULATION
inputs.nlpsol.nlpsolver='ess'; %add in solver here      % [] NLP solver: %n2fb
levenberg marquardt, dhc = direct hill climbing, no gradient, nl2sol and lsqnonlin is
levenberg marquardt
%
% LOCAL:
'local_fmincon'|'local_n2fb'|'local_dn2fb'|'local_dhc'|
%
'local_ipopt'|'local_solnp'|'local_nomad'|'local_nl2sol'|
%
'local_lsqnonlin'
%
MULTISTART:'multi_fmincon'|'multi_n2fb'|'multi_dn2fb'|'multi_dhc'|
%
'multi_ipopt'|'multi_solnp'|'multi_nomad'|'multi_nl2sol'|
%
'multi_lsqnonlin'
% GLOBAL: 'de'|'sres'
% HYBRID:
'hyb_de_fmincon'|'hyb_de_n2fb'|'hyb_de_dn2fb'|'hyb_de_dhc'|'hyb_de_ipopt'|
%
'hyb_de_solnp'|'hyb_de_nomad'|
%
'hyb_sres_fmincon'|'hyb_sres_n2fb'|'hyb_sres_dn2fb'|'hyb_sres_dhc'|
%
'hyp_sres_ipopt'|'hyb_sres_solnp'|'hyb_sres_nomad'
%
% METAHEURISTICS:
% 'ess' or 'eSS' (default)
% Note that the corresponding
defaults are in files:
% OPT_solvers\DE\de_options.m;
%
OPT_solvers\SS_**\ess_options.m

inputs.nlpsol.eSS.maxeval = 300000; %maximum number of evaluations
inputs.nlpsol.eSS.maxtime = 300; %max computational effort
inputs.nlpsol.eSS.log_var = [1:1:length(inputs.PEsol.global_theta_max)];
inputs.nlpsol.eSS.local.solver = 'lsqnonlin'; % 'nl2sol' uses gradient, lsqnonlin,
check that always converging to same solution
inputs.nlpsol.eSS.local.finish = 'fminsearch';

```


Bibliography

1. Hafner, A., Bulyk, M. L., Jambhekar, A., & Lahav, G. (2019). The multiple mechanisms that regulate p53 activity and cell fate. *Nature Reviews Molecular Cell Biology*, 20(4), 199-210.
2. Kasthuber, E. R., & Lowe, S. W. (2017). Putting p53 in context. *Cell*, 170(6), 1062-1078.
3. Chong, K. H., Samarasinghe, S., & Kulasiri, D. (2015). Mathematical modelling of p53 basal dynamics and DNA damage response. *Mathematical Biosciences*, 259, 27-42.
4. Haupt, Y., Maya, R., Kazaz, A., & Oren, M. (1997). Mdm2 promotes the rapid degradation of p53. *Nature*, 387(6630), 296-299.
5. Wade, M., Li, Y. C., & Wahl, G. M. (2013). MDM2, MDMX and p53 in oncogenesis and cancer therapy. *Nature Reviews Cancer*, 13(2), 83-96.
6. Ito, A., Lai, C. H., Zhao, X., Saito, S. I., Hamilton, M. H., Appella, E., & Yao, T. P. (2001). p300/CBP-mediated p53 acetylation is commonly induced by p53-activating agents and inhibited by MDM2. *The EMBO journal*, 20(6), 1331-1340.
7. Sabbatini, P., & McCormick, F. (2002). MDMX inhibits the p300/CBP-mediated acetylation of p53. *DNA and cell biology*, 21(7), 519-525.
8. Phillips, A., Teunisse, A., Lam, S., Lodder, K., Darley, M., Emaduddin, M., ... & Lenos, K. (2010). HDMX-L is expressed from a functional p53-responsive promoter in the first intron of the HDMX gene and participates in an autoregulatory feedback loop to control p53 activity. *Journal of Biological Chemistry*, 285(38), 29111-29127.
9. Lee, J. H., & Paull, T. T. (2005). ATM activation by DNA double-strand breaks through the Mre11-Rad50-Nbs1 complex. *Science*, 308(5721), 551-554.
10. Cheng, Q., & Chen, J. (2010). Mechanism of p53 stabilization by ATM after DNA damage. *Cell cycle*, 9(3), 472-478.

11. Batchelor, E., Loewer, A., & Lahav, G. (2009). The ups and downs of p53: understanding protein dynamics in single cells. *Nature Reviews Cancer*, 9(5), 371-377.
12. Lu, X., Ma, O., Nguyen, T. A., Jones, S. N., Oren, M., & Donehower, L. A. (2007). The Wip1 Phosphatase acts as a gatekeeper in the p53-Mdm2 autoregulatory loop. *Cancer cell*, 12(4), 342-354.
13. Lu, X., Nguyen, T. A., Zhang, X., & Donehower, L. A. (2008). The Wip1 phosphatase and Mdm2: cracking the "Wip" on p53 stability. *Cell Cycle*, 7(2), 164-168.
14. Zhang, X., Lin, L., Guo, H., Yang, J., Jones, S. N., Jochemsen, A., & Lu, X. (2009). Phosphorylation and degradation of MdmX is inhibited by Wip1 phosphatase in the DNA damage response. *Cancer research*, 69(20), 7960-7968.
15. Geva-Zatorsky, N., Rosenfeld, N., Itzkovitz, S., Milo, R., Sigal, A., Dekel, E., ... & Alon, U. (2006). Oscillations and variability in the p53 system. *Molecular systems biology*, 2(1).
16. Monk, N. A. (2003). Oscillatory expression of Hes1, p53, and NF- κ B driven by transcriptional time delays. *Current Biology*, 13(16), 1409-1413.
17. Batchelor, E., Mock, C. S., Bhan, I., Loewer, A., & Lahav, G. (2008). Recurrent initiation: a mechanism for triggering p53 pulses in response to DNA damage. *Molecular cell*, 30(3), 277-289.
18. Moenke, G., Cristiano, E., Finzel, A., Friedrich, D., Herzog, H., Falcke, M., & Loewer, A. (2017). Excitability in the p53 network mediates robust signaling with tunable activation thresholds in single cells. *Scientific reports*, 7, 46571.
19. Loewer, A., Batchelor, E., Gaglia, G., & Lahav, G. (2010). Basal dynamics of p53 reveal transcriptionally attenuated pulses in cycling cells. *Cell*, 142(1), 89-100.

20. Sin, C. (2012). *Algorithmic Parameter Space Reduction of a Systems Biology Model: A Case Study* (Master's thesis). Available from University of California, Los Angeles.
21. Greenwald W. (2016). *Modeling p53 Signaling Dynamics* (Master's thesis). Available from University of California, Los Angeles.
22. The Mathworks, Inc. (n.d.). *Matlab*. MathWorks. Retrieved December 18, 2021, from <https://www.mathworks.com/products/matlab.html>
23. Mathworks. (n.d.). *Estimate model parameters with parameter constraints (code)*. Estimate Model Parameters with Parameter Constraints (Code) - MATLAB & Simulink. Retrieved October 4, 2021, from <https://www.mathworks.com/help/sldo/ug/estimate-model-parameters-with-parameter-constraints-code.html>
24. The Mathworks, Inc. (n.d.). *Simulink - simulation and model-based design*. Mathworks. Retrieved December 18, 2021, from <https://www.mathworks.com/products/simulink.html>
25. Balsa-Canto, E., & Banga, J. R. (n.d.). *Amigo2 Toolbox*. Google Sites. Retrieved January 6, 2022, from <https://sites.google.com/site/amigo2toolbox/home>
26. The Mathworks, Inc. (n.d.). *How the Software Formulates Parameter Estimation as an Optimization Problem*. Mathworks. Retrieved December 22, 2021, from <https://www.mathworks.com/help/sldo/ug/optimization-problem-formulation-for-parameter-estimation.html>
27. DiStefano, J. J. (2013). 17 - Model Reduction and Network Inference in Dynamic Systems Biology. In *Dynamic Systems Biology Modeling and Simulation* (pp. 705–722). essay, Elsevier.
28. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. *Global Sensitivity Analysis. The Primer*, John Wiley and Sons, 2008.

29. The Mathworks, Inc. (n.d.). *What is Sensitivity Analysis?* Mathworks. Retrieved January 20, 2022, from <https://www.mathworks.com/help/sldo/ug/what-is-sensitivity-analysis.html>