# A dynamical low-rank approach to solve the chemical master equation for biological reaction networks

Martina Prugger[1,2], Lukas Einkemmer[3], Carlos F. Lopez [2,4]

**1** Department of Biochemistry, University of Innsbruck, Innsbruck, Tyrol, Austria
**2** Department of Biochemistry, Vanderbilt University School of Medicine, Nashville, TN, USA
**3** Department of Mathematics, University of Innsbruck, Innsbruck, Tyrol, Austria
**4** Department of Biomedical Informatics, Vanderbilt University Medical Center, Nashville, TN, USA

**Abstract**

Solving the chemical master equation is an indispensable tool in understanding the behavior of biological and chemical systems. In particular, it is increasingly recognized that commonly used ODE models are not able to capture the stochastic nature of many cellular processes. Solving the chemical master equation directly, however, suffers from the curse of dimensionality. That is, both memory and computational effort scale exponentially in the number of species. In this paper we propose a dynamical low-rank approach that enables the simulation of large biological networks. The approach is guided by partitioning the network into biological relevant subsets and thus avoids the use of single species basis functions that are known to give inaccurate results for biological systems. We use the proposed method to gain insight into the nature of asynchronous vs. synchronous updating in Boolean models and successfully simulate a 41 species apoptosis model on a standard desktop workstation.

Mechanistic models are commonly used to acquire insights about the biochemical reaction networks that govern cellular processes inside a cell. These models are typically obtained by a combination of expert and literature-driven knowledge as well as experimental data. Models based on ordinary differential equations (ODE) are most commonly used [1]. However, it is also well known that in a number of problems the ODE formulation is insufficient to describe important features of the biological system [2, 3, 4, 5]. Often this inability comes from the fact that ODE models replace stochastic dynamics by some average. In contrast, such stochastic effects are taken into account by models based on the chemical master equation (CME).

The primary challenge associated with models based on the CME is the large computational cost. For a direct solver, the computational effort and memory requirements increase exponentially with the number of species, limiting such simulations to relatively small systems. This is a reason why the less computationally expensive ODE models are popular. In this paper, we employ a dynamical low-rank approximation to drastically reduce the computational and memory costs and therefore make the solution of large, complex systems tractable. In our approach, the network is subdivided into subsets that are tightly coupled, while only averaged information is exchanged between them. This enables simulations with a network size that would be prohibitive for direct solvers. Although low-rank approximations have been used in the past — particularly in the quantum physics literature — our approach departs from traditional applications by exploiting the underlying structure of the network rather than reducing the equations to single species basis functions (orbitals in the language of quantum dynamics). In our view, this represents a paradigm shift of how these methods should be applied to obtain accurate results with drastically reduced computational effort.

We note that in the literature the large computational cost of solving the CME has been addressed in a variety of ways. If a system is very sparse (i.e. only a few chemical states are populated) finite state projection [6] is a viable way to reduce computational complexity. However, especially for Boolean models (for details see below) this essentially assumes that a large number of species is either on or off with probability 1, a very unrealistic assumption. The other commonly used technique is Monte-Carlo simulation or SSA [7, 8]. In this approach trajectories in the space of possible configurations are sampled and a statistical representation is obtained by repeated simulations. Thus, to obtain, e.g., low abundant phenotypes requires the computation of an excessively large number (depending on the problem potentially tens to hundreds of thousands) of trajectories [9].

The idea of dynamical low-rank approximation is to use lower dimensional basis functions in order to approximate the high dimensional probability distribution. If a relatively small number of such basis functions is sufficient to describe the relevant dynamics, this results in a drastic reduction of the required computational resources. In [10], such an approach is presented for a relatively low-dimensional CME. In this work, following the quantum physics literature, each species is considered its own sub-

space (i.e. the basis functions only depend on a single species). The proposed method works relatively well for the given examples ($\approx$ five species and $\approx$ ten reactions), but has the disadvantage that an approximation error is made in each reaction. In biological applications, it would be difficult or even impossible to consider each species as independent, particularly given the intricate structures of complex biological networks [11]. Therefore, approximating the probability distribution of a system of reactions using a single species basis function could significantly degrade the accuracy of the method. For this reason, single basis functions have been considered as poor representations for biological systems.

Our proposed dynamical low-rank approximation formalism for biological networks leverages prior work. In particular, the projector splitting based dynamical low-rank approach introduced in [12] and its later extension to hierarchical tensors [13]. This was the first robust low-rank approach able to handle dynamical systems [14]. More recently, this approach found success in solving particle kinetic equations such as the Vlasov equation [15, 16, 17] and radiative transfer problems [18, 19, 20, 21]. Our dynamical low-rank approach is fully deterministic (i.e. noise free) and provides the entire probability distribution with a single simulation. This enables us to get an accurate picture of the network dynamics, while still being relatively cheap from a computational point of view. In addition, the proposed approach does not show unfavorable scaling with the number of reactions (as does SSA), but only with the dependent species for each reaction. The approximation error in the dynamical low-rank approach is controlled by how the network is partitioned, thus allowing us to make use of expert knowledge to refine the result.

While the dynamical low-rank approach can be applied to any model based on the CME, in this paper we consider Boolean models [22]. In this formalism, each biochemical species (e.g. genes, proteins) can exist in two states (on or off), which significantly simplifies model complexity and only requires that one knows the structure of the network (which in some situations can be inferred from data [23]). The state of the network is then determined by a binary string that encodes which species are switched on and which are switched off. In the CME approach a probability is assigned to each such state. Boolean models have proven very useful to extract mechanistic insights from complex systems [24, 25] and a range of software packages are available for practitioners. We note that the Boolean model formalism is also used in a variety of applications outside biology including circuit theory, natural language processing, and search engines [26, 27]. For increasingly large and complex problems Boolean models have the added advantage that they do not require precise knowledge of a large number of kinetic parameters, which otherwise must be either measured experimentally, or inferred using statistical methods [28, 29].

## Results

### An illustrative example: a Boolean model of the mTOR pathway

To demonstrate the use of the low-rank approximation based on network structure for a biochemical reaction network, we chose the model of the mTOR pathway introduced in [30]. The model includes 22 species and its network diagram is given in Figure 1. The full Boolean ruleset can be found in the file **supplement2_mTOR.pdf**.

#### Network partitioning

The first step in our dynamical low-rank approach requires the partitioning of the network into subnetworks while leveraging the network structure. For this step, we require that the partitioning of the network yields subnetworks with an approximately equal number of species. Every interaction that remains within its partition (e.g. the rule for *TSC1/2* in the blue partitioning in Figure 1) is treated exactly by the algorithm. Therefore, we want to minimize cross-partition dependencies (indicated by arrows in the Figure) as these interactions will be subject to low-rank approximations (e.g. the pathway that connects *TSC1/2* to *RHEB* in the blue partitioning). In Figure 1, we introduce five different partitionings that we will use in the following. The blue, green, purple, and red networks are chosen following this general principle of minimizing connections between the subnetworks. To demonstrate the consequences of a bad partitioning, we also included the yellow partition, where the species are randomly assigned.

The most important parameter of the low-rank approximation is the choice of rank. A smaller rank $r$ leads to faster compute time and less memory requirements for a simulation. However, decreasing rank can also lead to larger error. The maximum rank of a simulation is given by $2^{d/2}$, where $d$ is the number of chemical species. In our example, d=22 and thus the maximum possible rank would be 2048. At the highest rank, the simulation is exact but this does not result in any computational improvements in compute time or memory use relative to a direct solver. In Figure 2 (top-left), we compare the errors of five different network partitions in relation to the exact solution for different ranks (namely, $r = 2$, $r = 4$, $r = 8$, and $r = 16$). It is assumed that initially all states of the system are equally likely. For the blue (reference partition scheme) we can see that in order to obtain an error on the order of 1% (0.01) only rank 4 is required. To reduce this error to $10^{-3}$ the rank has to be increased to approximately 16. We also clearly see from the top-left of Figure 2, by comparing e.g. the blue with the yellow partitioning, that choosing an appropriate partitioning can

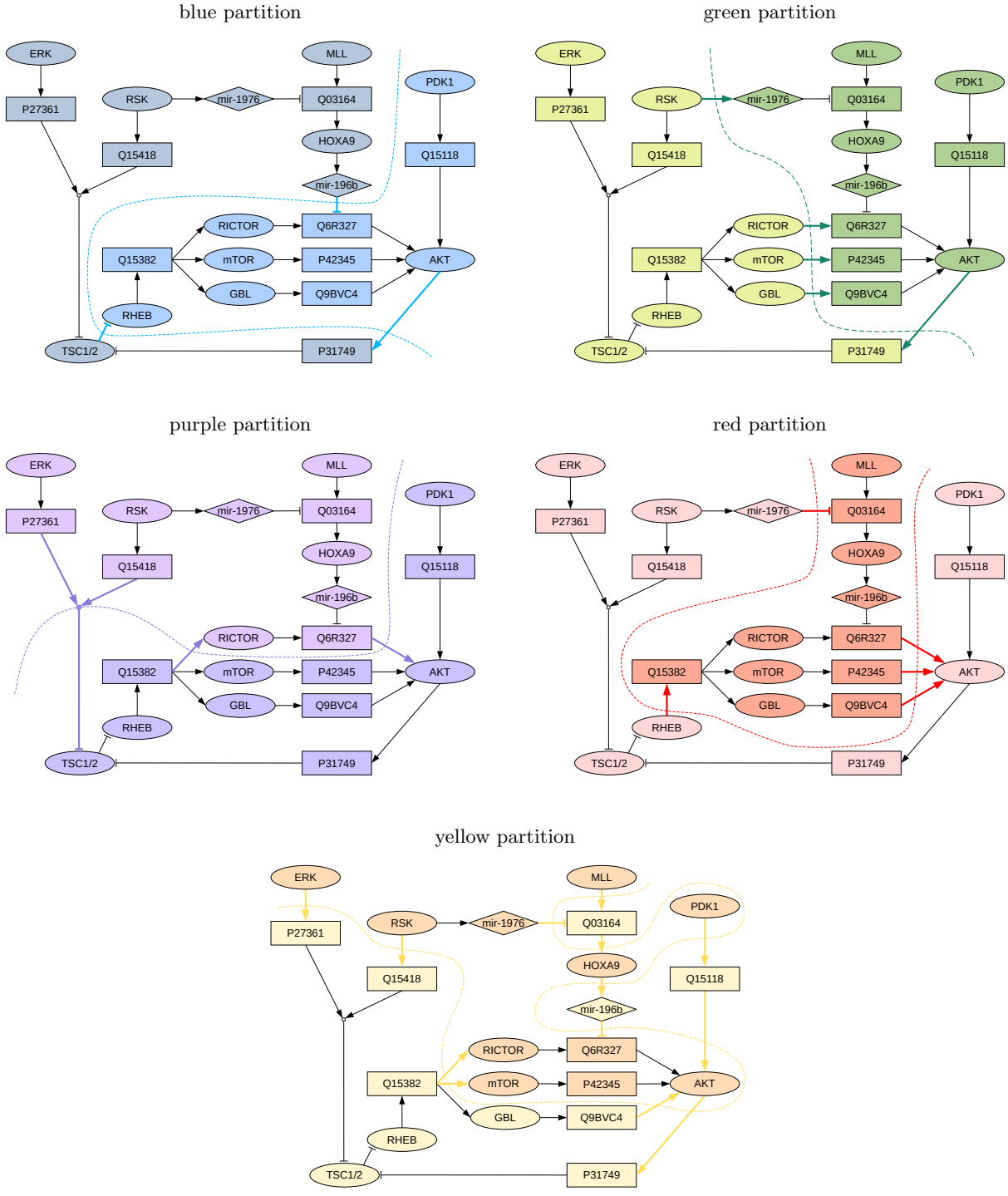


Figure 1: Boolean networks for the mTOR pathway model. We consider five different partitionings for the low-rank algorithms (blue, green, purple, red, and yellow). How the partitioning is done influences how small the rank (which is proportional to computational cost) can be chosen for a given accuracy. The blue and purple partitioning perform best as they treat the important pathway from *Q15382* to *AKT* with minimal error.

significantly reduce the rank and thus computational effort. Since we chose the yellow network randomly, we do not expect the partitioning to be competitive. In fact, as we can see, even increasing the rank to 16 does not decrease the error below its initial threshold, while the other partitionings see a significant reduction in the error. We also note that minimizing the connections between two partitions is not enough information in order to obtain an ideal partitioning. This can be seen in the behavior of the red and the green network. Both networks have five edges that separate the two partitions, but their accuracy is very different. The reason for this is that both the green and the red network disturb the important pathway from *Q15382* to *AKT* in a significant way.

### Steady states and dynamics of mTOR

The proposed dynamical low-rank integrator (see Methods for the algorithm) is able to obtain all the statistical information in a single simulation because it is directly solving the chemical master equation. The results of the corresponding simulation are shown in in Fig 2 (top-right). We display only the steady states that have a probability that exceeds $10^{-3}$ (20 in total for this model). For very small ranks the approximation becomes inaccurate and SS9 and SS10 are lost. However, even for $r = 5$ all of the relevant steady states are captured and starting at rank $r = 10$ the probabilities match exactly those of the reference solution.

In addition to the steady states, the dynamical low-rank algorithm also provides information about which trajectories lead to a given steady states as well as their probability. In order to study such time-dependent effects the dynamical low-rank approach is particularly useful. E.g. Consider the dynamics in Figure 3, where initially only a few states have a non-zero probability. Moreover, after some time the system settles in a small number of steady states. Both of these would be well represented by finite state projection. However, in the dynamics that the system follows to reach these few steady states we see an explosion in the number of states with non-zero probability. The dynamical low-rank approach, in contrast to finite state projection (which here again would suffer from the curse of dimensionality), is able to resolve these intermediate states without any increase in computational cost as we can see from Figure 3.

Finally, we look at the dynamics predicted by the dynamical low-rank algorithm. In Figure 2 bottom we plot the time evolution of *mTOR* for the different partition schemes and different ranks for the uniform initial condition (left) and the single initial condition (right). We observe that the blue and purple partitions in particular give results in agreement with the exact solution, even for rank 4. For rank 16 all partition schemes display the correct behavior.

We conclude that for appropriately chosen partitionings even a small rank (i.e. low computational cost) is sufficient to obtain accurate results (both in terms of steady states and dynamics). If, for some reason, a good partitioning is not available, the rank needs to be increased which adds to the computational and memory cost.

### Computational efficiency

The mTOR network described in some detail in the previous section with its 22 species is still a relatively small model. Ultimately, we are interested in using the dynamical low-rank approach for larger problems. In the subsequent sections we will discuss two examples. A model of pancreatic cancer with 34 species and a more detailed model of apoptosis with 41 species. To illustrate the dramatic growth of memory requirements that results from increasing the number of species let us assume that a direct solver would only need to store a single input and output vector (a highly questionable assumption as a real implementation would most likely significantly exceed this theoretical value). In this case, the pancreatic cancer model would require 275 GB of main memory (RAM). This renders it beyond the scope of a desktop workstations and would require parallelization to a distributed memory supercomputer. For the apoptosis model considered the requirements would be even more severe. In fact, 35 TB of RAM would be required in this case, which is clearly not feasible.

In Table 1 (top) we show that our C++ implementation of the low-rank algorithm drastically reduces these requirements. For the pancreatic cancer problem running simulations with $r = 10$, which gives very accurate results (see **supplement2_mTOR.pdf**), only 421 MB are required which is a reduction by a factor of 650. For the apoptosis model this is even more pronounced, as we would expect. In this case 12 GB are required for $r = 10$, a reduction by a factor of approximately 3000. Together with the timings provided in Table 1 (bottom) this clearly demonstrates that using the dynamical low-rank approach proposed here enables us to easily conduct these simulations on a desktop workstation within a reasonable amount of wall clock time.

### Asynchronous vs. synchronous updating for a pancreatic cancer model

Results using the stochastic simulation algorithm (SSA) can be obtained using two distinct update rules. In synchronous updating all rules are applied at the same time and thus all species are updated simultaneously. While synchronous updating is commonly used, there is significant debate as to its biological interpretation [31, 32, 33, 34]. Asynchronous updating has been proposed as an alternative [35, 36, 37, 38, 39]. In asynchronous updating a single reaction is selected at random and then the corresponding rule is applied. As such, it has a direct interpretation in terms of chemical reactions and it has been shown that asynchronous updated SSA in the limit

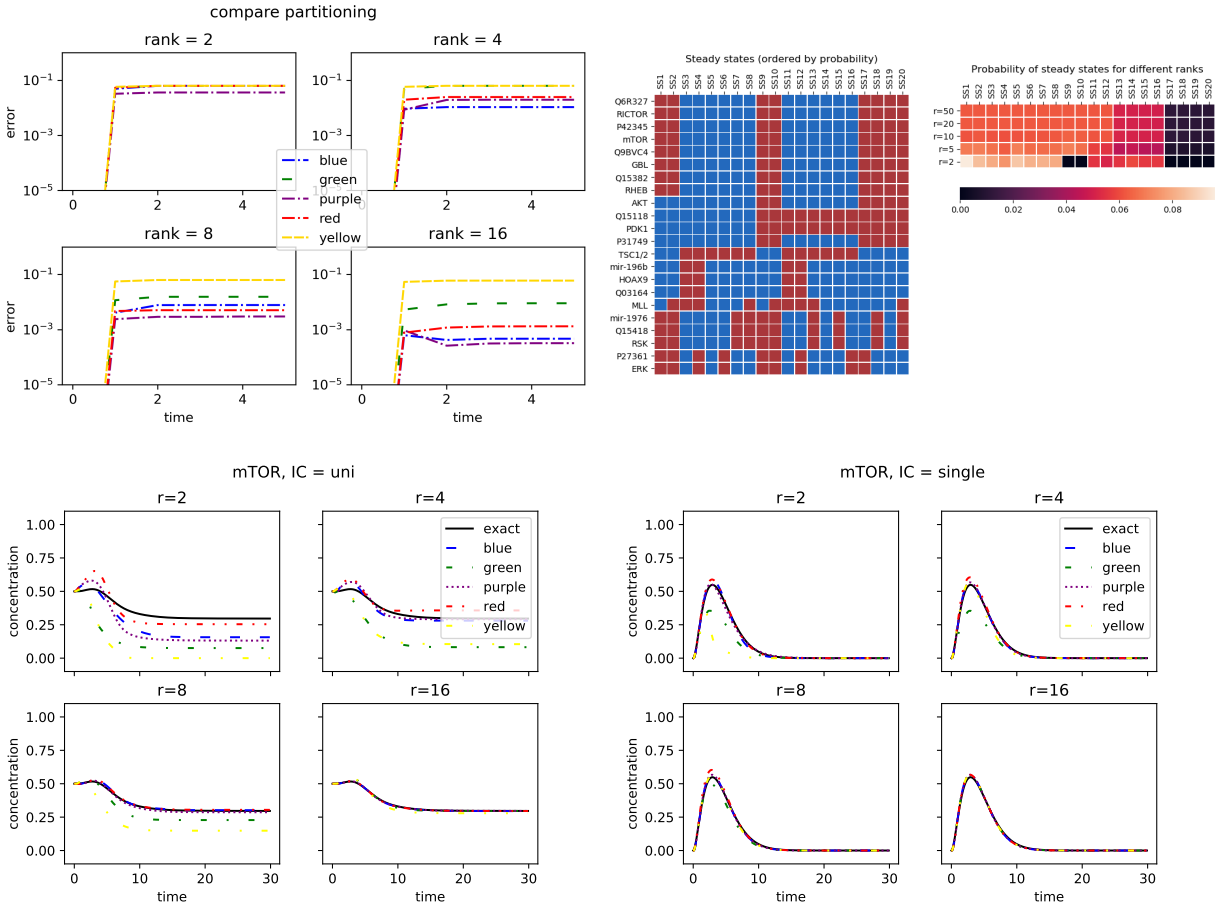


Figure 2: On the top-left the error of the dynamical low-rank approximation for the five different partitionings of the mTOR network is shown for ranks 2, 4, 8, and 16. The error is the maximum of the difference in probability between the low-rank approximation and the exact solution at a given time. On the top-right the steady states with probability larger than $10^{-3}$ are shown, sorted by their respective probability (red indicates on, blue indicates off). The blue partitioning and an initial value in which all states are equally likely are used. On the bottom the time evolution of the probability that $mTOR$ is switched on is shown. The configuration on the left starts with an initial configuration where $mTOR$ initially is equally likely to be in an on or an off state. The configuration on the bottom has $mTOR$ initially switched off and after a transient phase that is mediated by a significant concentration of $mTOR$ results in a steady state where $mTOR$ is switched off again (with high probability).

of a large number of samples gives the same dynamics as the chemical master equation [40, 41].

Nevertheless, it has been argued that synchronous updating is sufficient in order to obtain the relevant steady states [42, 43, 44, 45, 46]. However, recent work, performed using direct and SSA simulation, has called this assertion into question [47, 48]. Our dynamical low-rank approach gives us a way to easily investigate the full dynamics of large systems in this context. As it is based on the chemical master equation, the dynamical low-rank approach encodes the biologically relevant asynchronous updating, while allowing us to obtain statistics over a range of trajectories and initial values with a single run of the simulation.

We illustrate this for pancreatic cancer. Studies have found that overexpression of *HMGB1* can lead to increased proliferation and decreased apoptosis in cancer cells [49]. To model this, a single-cell Boolean network was introduced in [50]. In this model *HMGB1* acts as a parameter that influences the ultimate fate of the cell. Since the proteins *P53*, *RAS*, and *INK4a* are either mutated or lost in pancreatic cancer (with a high probability), we will only look at initial values with *P53* and *INK4a* set to off and *RAS* set to on. The suggested network partitionings as well as additional simulation results can be found in **supplement3_pancreatic.pdf**.

| problem | $r = 2$ | $r = 5$ | $r = 10$ | $r = 20$ | $r = 30$ | $r = 50$ | direct simulation |
|---|---|---|---|---|---|---|---|
| mTOR | 34 MB | 38 MB | 83 MB | 191 MB | – | 113 MB | > 66 MB |
| pancreatic | 160 MB | 267 MB | 421 MB | 671 MB | – | 1.4 GB | > 275 GB |
| apoptosis | 2.5 GB | 6.1 GB | 12 GB | 24 GB | 36 GB | – | > 35 TB |

| problem | $r = 2$ | $r = 5$ | $r = 10$ | $r = 20$ | $r = 30$ | $r = 50$ |
|---|---|---|---|---|---|---|
| mTOR | 0.6 s | 1.1 s | 3.4 s | 4.3 s | – | 10 s |
| pancreatic | 1 m 39 s | 2 m 55 s | 5 m 37 s | 9 m 08 s | – | 28 m 47 s |
| apoptosis | 13 m 08 s | 33 m 21 s | 1 h 06 m 33 s | 1 h 39 m 06 s | 2 h 39 m 35 s | – |

Table 1: Main memory (RAM) requirements for the mTOR pathway model (22 species), pancreatic cancer (34 species), and apoptosis (41 species) for the dynamical low-rank algorithm and different values of the rank $r$ is shown on the top. For comparison, a theoretical lower bound for the memory that a direct solver would require is also provided (note, however, that an actual implementation will most likely significantly exceed this value). In all simulations the reference partitioning (blue and setup a, respectively) is used. On the bottom, timing measurements for the dynamical low-rank algorithm and different values of the rank $r$ are presented. The problem is integrated until a final time $T = 1$ and all simulations have been run on a Intel Core i9-7940X CPU with 14 cores and 64 GB of DDR4 memory.

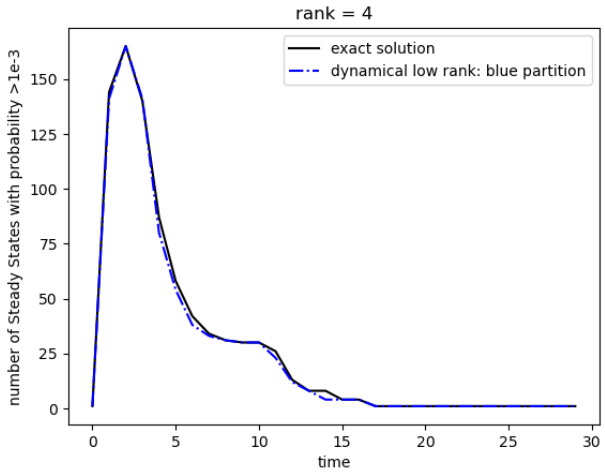


Figure 3: The number of states with probability larger than $10^{-3}$ for the mTOR network is shown as a function of time. Initially we start with a deterministic initial value (which is rank 1) and ultimately we end up in a single steady state with high probability (again rank 1). However, a large number of trajectories is generated intermittently in this dynamics.

In [50] mathematical techniques were used in order to prove that certain properties of the Boolean network are compatible with the experimental observations. For example, it is shown that overexpression of *HMGB1* will necessarily result in steady states with *Proliferate* switched on and *Apoptosis* switched off. However, these results only apply to synchronous updating. The dynamical low-rank approach, on the other hand, allows us to investigate this for the chemical master equation (i.e. with asynchronous updating). The time evolution of the probability that the cell is in the *Proliferate* state or that *Apoptosis* was triggered is shown in Figure 4. Note that we conclude from these results that in the asynchronous case the dynamics are significantly more complicated. It is possible, that apoptosis is triggered if *HMGB1* is turned on (the steady-state probability is approximately 16%). Moreover, there is no difference in the ultimate cell fate (i.e. with respect to proliferation or apoptosis) whether *HMGB1* is switched on or off. Thus, there is a significant difference between the steady states in the asynchronous case compared to the synchronous case that is discussed in [50]. This casts significant doubt that the model (at least in the asynchronous case) can describe the experimentally observed effects accurately. This also highlights that models derived in the synchronous case can not (at least not without alteration) be considered a viable description of the relevant biology.

Our results show that even if one is only interested in the steady states, the result for synchronous and asynchronous updating may exhibit significant differences. The dynamical low-rank approach is a tool particularly well suited for obtaining steady states that are consistent with the biologically more relevant asynchronous updating. The most likely steady states are shown in Figure 4. We once again highlight the fact that even a small rank (e.g. $r = 5$) is sufficient to capture the steady states as well as their probability very well.

## Low-rank approximation and cell-death execution

To demonstrate the use of a low-rank approximation in a real-world problem, we now consider a Boolean network describing programmed cell death regulation taken from [51]. In this model, competing signals for death and growth and their interactions are modeled to explore the

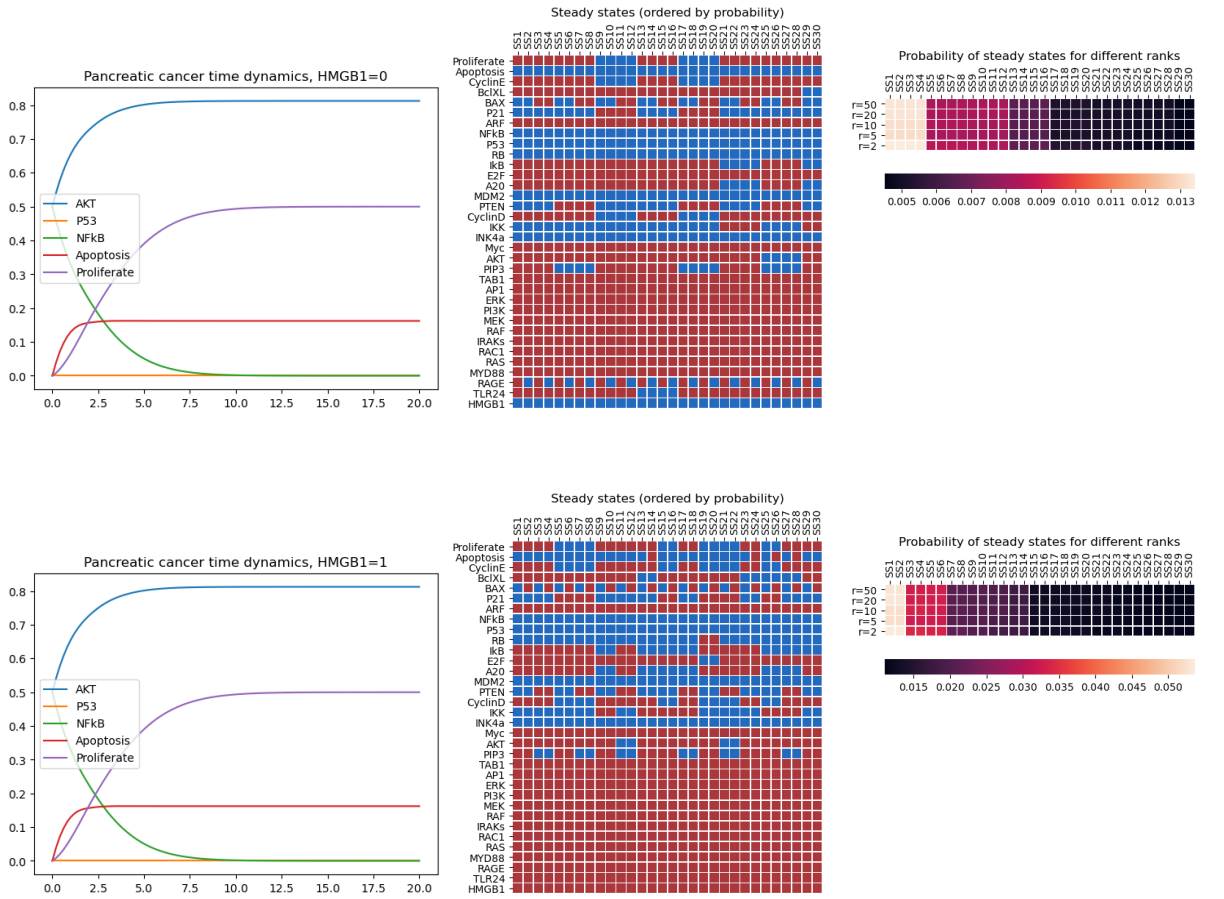


Figure 4: The dynamics of selected species for the pancreatic cancer model (left) and the steady sates (right) are shown for *HMGB1* switched off (top) and *HMGB1* switched on (bottom). Steady states are sorted according to their probability. Red represents that the species is on, blue represents that the species in this steady state is off. Note, that the dynamics for *Akt*, *p53*, *NFkB*, *Apoptosis*, and *Proliferate* are exactly the same. The number of steady states, however, are much reduced for *HMGB=1* (the first couple of significant steady states have a much higher probability than for *HMGB=0*).

mechanisms that ultimately determine the fate of the cell. The model inputs are either a signal representing Tumor Necrosis Factor (TNF) or a growth factor (GF) meant to capture stimulus by a ligand such as Epidermal Growth Factor. The main observation in the original paper, is that setting *GF* to "on" is ineffective in reducing apoptosis in isolation, but can reduce apoptosis execution when combined with a *TNF* signal. This behavior can also be observed in the simulation results presented in Figure 5. We therefore use this model as a reference model to demonstrate the use of the dynamic low-rank approximation method introduced in the previous section.

As shown in Table 1, direct evaluation of the chemical master equation with traditional methods would require 35 Terabytes of RAM and immense amounts of computer time. However, dynamic low-rank approximation of the problem with various ranks converts the problem from intractable to easily accessible with modest computation resources on a modern desktop or laptop computer.

Figure 5 shows the dynamics of *DNAdam* (which in this model is a proxy for programmed cell death) alongside five of the important proteins/genes. The full list of species dynamics can be found in **supplement4_apoptosis.pdf**. With *TNF* switched off the probability of *Apoptosis* is approximately 60%. This is independent whether *GF* is on or off. Switching *TNF* on increases the likelihood of apoptosis dramatically. In this setting switching on *GF* significantly suppresses apoptosis. This is consistent with

previous results from the literature [51]. The mechanism of the network can be understood from Figure 5 as follows: *TNF* directly affects a cascade that includes various caspases that via *Cas3* as the last link in the chain lead to *Apoptosis*. If this cascade is active *AKT* (which is downstream of *GF*) can act as an inhibitor to *Cas9* breaking the direct link between *TNF* and *Cas3*. Thus, if the caspase cascade is active *GF* is able to suppress it. It has, however, no significant effect on the baseline *Apoptosis* rate (which is mediated via *p53*) and does not rely on the caspase cascade.

## Discussion

We have shown that our dynamical low-rank approach based on partitioning the network into biological relevant parts can achieve accurate results for the CME both in terms of dynamics and steady state results. Since a small rank (usually between $r = 5$ and $r = 15$) is sufficient, this approach results in a drastic reduction in memory and consequently allows us to solve large problems (such as the apoptosis model with 41 species for which a direct simulation would clearly be prohibitive) on a typical desktop workstation. We have also used our method to gain insight into the nature of synchronous vs. asynchronous updating, which is still heavily debated in the community. Our work shows that for the studied pancreatic cancer model neither the steady states nor the dynamics under synchronous updating reflects well the observations made by solving the chemical master equation (i.e. asynchronous updating).

The most commonly used technique to address the large computational cost of solving the chemical master equation is Monte-Carlo simulation or SSA. In this approach trajectories in the space of possible configurations are sampled and a statistical picture is obtained by repeated simulations. Thus, to obtain, e.g., low abundant phenotypes requires the computation of an excessively large number of trajectories. In contrast, our dynamical low-rank approach is fully deterministic (i.e. noise free) and provides the entire probability distribution with a single simulation. This enables us to obtain an accurate picture of the network dynamics and steady state behavior, while still being relatively cheap from a computational point of view. In addition, the proposed approach does not show unfavorable scaling with the number of reactions (as does SSA), but only with the dependent species for each reaction (usually a small number). The approximation error in the dynamical low-rank approach is controlled by how the network is partitioned, thus allowing us to make use of expert knowledge to refine the result. A possible alternative would be to use graph algorithms in order to perform the partitioning automatically. This will be the subject of future work.

## Methods

### Dynamical low-rank approximation

The chemical master equation describes the dynamics (i.e. time evolution) of a probability distribution $P(t, x_1, \dots, x_d)$. In the case of a Boolean model we have $x_i \in \{0, 1\}$. For example, for the mTOR network (see Figure 1 for details) $x_1$ specifies whether *ERK* is turned on ($x_1 = 1$) or off ($x_1 = 0$). The value of $P(t, x_1, \dots, x_d)$ specifies the probability that our system is in the state $(\text{ERK}, \dots) = (x_1, \dots, x_d)$ at time $t$.

The first step in the low-rank algorithm is to partition the set of species into two groups $\{x_1, \dots, x_m\}$ and $\{x_{m+1}, \dots, x_d\}$ with $m \approx d/2$. We then write the probability density function in the following matrix form (assuming a fixed $t$ and suppressing the explicit time dependence). For example, for a network with four species the probability density function is thus written in the following form

$$P = \begin{bmatrix} P(0,0,0,0) & P(1,0,0,0) & P(0,1,0,0) & P(1,1,0,0) \\ P(0,0,1,0) & P(1,0,1,0) & P(0,1,1,0) & P(1,1,1,0) \\ P(0,0,0,1) & P(1,0,0,1) & P(0,1,0,1) & P(1,1,0,1) \\ P(0,0,1,1) & P(1,0,1,1) & P(0,1,1,1) & P(1,1,1,1) \end{bmatrix}.$$

This directly encodes the partitioning. The states with different $x_1$ and $x_2$ are put in the same row and states with different $x_3$ and $x_4$ are put in the same column. More details can be found in **supplement1_lowrank.pdf**.

Storing the $2^m \times 2^{d-m}$ matrix $P$ still requires $2^d$ entries, which is prohibitive from both a memory and computational point of view. To reduce the degrees of freedom required, we use the low-rank approximation

$$P = USV^T,$$

where $U$ is a $2^m \times r$, $V$ is a $2^{d-m} \times r$, and $S$ is a $r \times r$ matrix. If the rank $r$ is small, as is the case in many problems as we have shown, then storing the low-rank factors $U$, $V$, $S$ takes significantly less memory compared to storing $P$. The essential idea of the low-rank approximation is to describe the state of the system with the low-rank factors only and to never form the full matrix $P$.

The chemical master equation is a dynamical system and it is thus necessary to find a way to determine the time evolution of the low-rank factors. Our approach is based on early work in the quantum physics literature, see e.g. [14, 52]. For a mathematical treatment see e.g. [53, 54]. In the seminal paper [12] the projector splitting integrator was introduced, which allows for a robust treatment of the evolution equations associated for the low-rank factors (i.e. no regularization is required). We outline this approach in Algorithm 1

The dynamical low-rank algorithm can be applied to a range of problems. The information on the specific problem to be solved enters in the update $\Delta P$ (the orange marked parts of Algorithm 1). The terms $(\Delta P)V_0$, $U_1^T \Delta P$, etc. describe the projection of the update on the tangent space of the approximation manifold. It is *crucial* that orange-colored routines of the algorithm are computed efficiently. Determining $\Delta P$ and then multiplying

**Algorithm 1:** Projector splitting dynamical low-rank algorithm. In each time step the low-rank factors are updated according to the described algorithm.

**Input:** $U_0$, $S_0$, $V_0$
**Output:** $U_1$, $S_1$, $V_1$

$K_1 = U_0 S_0 + (\Delta P) V_0$;
QR decomposition to get $U_1 \hat{S}_1 = K_1$;
$\tilde{S}_0 = \tilde{S}_1 - U_1^T (\Delta P) V_0$;
$L_1 = V_0 \tilde{S}_0^T + (U_1^T \Delta P)^T$;
QR decomposition to get $V_1 S_1^T = L_1$

with $V_0$, for example, would require assembling the matrix $\Delta P$ which is prohibitive. The main insight required in order to do this efficiently is that for the chemical master equation each reaction can be treated independently and usually only depends on a small number of species. More details are provided in **supplement1_lowrank.pdf**.

## Code availability & validation

The code used to run the simulation and generate the plots is available at `https://bitbucket.org/mprugger/low_rank_cme`.

We have compared the results obtained with the dynamical low-rank implementation with a direct simulation method for a range of smaller networks (including the mTOR pathway model considered in this paper). In addition, as part of the code, a number of automated unit tests are available.

## Acknowledgements

This work was supported by National Science Foundation (NSF) CAREER award MCB 1942255 to CFL, National Institutes of Health (NIH) award 1U01CA215845 to CFL (M-PI).

## References

[1] W. W. Chen, M. Niepel, and P. K. Sorger. Classic and contemporary approaches to modeling biochemical reactions. *Genes & development*, 24(17):1861–1875, 2010.

[2] M. K. Tonn, P. Thomas, M. Barahona, and D. A. Oyarzún. Stochastic modelling reveals mechanisms of metabolic heterogeneity. *Communications biology*, 2(1):1–9, 2019.

[3] S. Grima and S. Schnell. Modelling reaction kinetics inside cells. *Essays in Biochemistry*, 45:41–56, 2008.

[4] M. Niepel, S. L. Spencer, and P. K. Sorger. Non-genetic cell-to-cell variability and the consequences for pharmacology. *Current opinion in chemical biology*, 13(5-6):556–561, 2009.

[5] P. Paszek, S. Ryan, L. Ashall, K. Sillitoe, C. V. Harper, D. G. Spiller, D. A. Rand, and M. RH. White. Population robustness arising from cellular heterogeneity. *Proceedings of the National Academy of Sciences*, 107(25):11644–11649, 2010.

[6] B. Munsky and M. Khammash. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of chemical physics*, 124(4):044104, 2006.

[7] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of computational physics*, 22(4):403–434, 1976.

[8] L. A. Harris and P. Clancy. A "partitioned leaping" approach for multiscale modeling of chemical reaction dynamics. *The Journal of chemical physics*, 125(14):144107, 2006.

[9] D. T. Gillespie, A. Hellander, and L. R. Petzold. Perspective: Stochastic algorithms for chemical kinetics. *The Journal of chemical physics*, 138(17):05B201_1, 2013.

[10] T. Jahnke and W. Huisinga. A dynamical low-rank approach to the chemical master equation. *Bulletin of mathematical biology*, 70(8):2283, 2008.

[11] A.-L. Barabási. Scale-Free Networks: A Decade and Beyond. *Science*, 325(5939):412–413, 2009.

[12] C. Lubich and I.V. Oseledets. A projector-splitting integrator for dynamical low-rank approximation. *BIT Numer. Math.*, 54(1):171–188, 2014.

[13] C. Lubich, T. Rohwedder, R. Schneider, and B. Vandereycken. Dynamical approximation by hierarchical Tucker and tensor-train tensors. *SIAM Journal on Matrix Analysis and Applications*, 34(2):470–494, 2013.

[14] H.-D. Meyer, U. Manthe, and L. S. Cederbaum. The multi-configurational time-dependent Hartree approach. *Chemical Physics Letters*, 165(1):73–78, 1990.

[15] L. Einkemmer and C. Lubich. A low-rank projector-splitting integrator for the Vlasov–Poisson equation. *SIAM Journal on Scientific Computing*, 40(5):B1330–B1360, 2018.

[16] L. Einkemmer, A. Ostermann, and C. Piazzola. A low-rank projector-splitting integrator for the Vlasov–Maxwell equations with divergence correction. *Journal of Computational Physics*, 403:109063, 2020.

[17] L. Einkemmer and I. Joseph. A mass, momentum, and energy conservative dynamical low-rank scheme for the Vlasov equation. *Journal of Computational Physics*, page 110495, 2021.

[18] Z. Peng, R. G. McClarren, and M. Frank. A low-rank method for two-dimensional time-dependent radiation transport calculations. *Journal of Computational Physics*, 421:109735, 2020.

[19] L. Einkemmer, J. Hu, and Y. Wang. An asymptotic-preserving dynamical low-rank method for the multi-scale multi-dimensional linear transport equation. *Journal of Computational Physics*, 439:110353, 2021.

[20] Z. Ding, L. Einkemmer, and Q. Li. Dynamical low-rank integrator for the linear Boltzmann equation: error analysis in the diffusion limit. *SIAM Journal on Numerical Analysis*, 59(4):2254–2285, 2021.

[21] L. Einkemmer, J. Hu, and L. Ying. An Efficient Dynamical Low-Rank Algorithm for the Boltzmann–BGK Equation Close to the Compressible Viscous Flow Regime. *SIAM Journal on Scientific Computing*, 43(5), 2021.

[22] J. GT. Zañudo, S. N. Steinway, and R. Albert. Discrete dynamic network modeling of oncogenic signaling: Mechanistic insights for personalized treatment of cancer. *Current Opinion in Systems Biology*, 9:1–10, 2018.

[23] M. Prugger, L. Einkemmer, S. P. Beik, P. T. Wasdin, L. A. Harris, and C. F. Lopez. Unsupervised logic-based mechanism inference for network-driven biological processes. *PLOS Computational Biology*, 17(6):e1009035, 2021.

[24] A. Yachie-Kinoshita, K. Onishi, J. Ostblom, M. A. Langley, E. Posfai, J. Rossant, and P. W. Zandstra. Modeling signaling-dependent pluripotency with Boolean logic to predict cell fate transitions. *Molecular systems biology*, 14(1):e7952, 2018.

[25] M. A. Clarke and J. Fisher. Executable cancer models: successes and challenges. *Nature Reviews Cancer*, 20(6):343–354, 2020.

[26] L. Null and J. Lobur. *Essentials of Computer Organization and Architecture*. Jones & Bartlett Learning, 2018.

[27] J. Barwise, J. Etchemendy, G. Allwein, D. Barker-Plummer, and A. Liu. *Language, proof and logic*. CSLI publications, 2000.

[28] E. M. Shockley, J. A. Vrugt, and C. F. Lopez. PyDREAM: high-dimensional parameter inference for biological models in python. *Bioinformatics*, 34(4):695–697, 2018.

[29] A. C. Babtie, T. E. Chan, and M. PH. Stumpf. Learning regulatory models for cell development from single cell transcriptomic data. *Current Opinion in Systems Biology*, 5:72–81, 2017.

[30] A. Benso, S. Di Carlo, G. Politano, A. Savino, and A. Vasciaveo. An extended gene protein/products boolean network model including post-transcriptional regulation. *Theoretical Biology and Medical Modelling*, 11(1):1–17, 2014.

[31] S. Barman and Y.-K. Kwon. A boolean network inference from time-series gene expression data using a genetic algorithm. *Bioinformatics*, 34(17):i927–i933, 2018.

[32] S. Gao, C. Xiang, C. Sun, K. Qin, and T. H. Lee. Efficient boolean modeling of gene regulatory networks via random forest based feature selection and best-fit extension. In *2018 IEEE 14th International Conference on Control and Automation (ICCA)*, pages 1076–1081. IEEE, 2018.

[33] S. Martin, Z. Zhang, A. Martino, and J.-L. Faulon. Boolean dynamics of genetic regulatory networks inferred from microarray time series data. *Bioinformatics*, 23(7):866–874, 2007.

[34] N. Shi, Z. Zhu, K. Tang, D. Parker, and S. He. Aten: And/or tree ensemble for inferring accurate boolean network topology and dynamics. *Bioinformatics*, 36(2):578–585, 2020.

[35] L. Paulevé, J. Kolčák, T. Chatain, and S. Haar. Reconciling qualitative, abstract, and scalable modeling of biological networks. *Nature communications*, 11(1):1–7, 2020.

[36] J. J. Tyson, K. C. Chen, and B. Novak. Network dynamics and cell physiology. *Nature reviews Molecular cell biology*, 2(12):908–916, 2001.

[37] J. J. Tyson, K. C. Chen, and B. Novak. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Current opinion in cell biology*, 15(2):221–231, 2003.

[38] M. L. Wynn, N. Consul, S. D. Merajver, and S. Schnell. Logic-based models in systems biology: a predictive and parameter-free network analysis method. *Integrative biology*, 4(11):1323–1337, 2012.

[39] J. GT. Zanudo and R. Albert. Cell fate reprogramming by control of intracellular network dynamics. *PLoS computational biology*, 11(4):e1004193, 2015.

[40] A. Ribeiro, R. Zhu, and S. A. Kauffman. A general modeling strategy for gene regulatory networks with stochastic dynamics. *Journal of computational Biology*, 13(9):1630–1639, 2006.

[41] G. Stoll, E. Viara, E. Barillot, and L. Calzone. Continuous time boolean modeling for biological signaling: application of gillespie algorithm. *BMC systems biology*, 6(1):1–18, 2012.

[42] J. D. Schwab, S. D. Kühlwein, N. Ikonomi, M. Kühl, and H. A. Kestler. Concepts in boolean network modeling: What do they all mean? *Computational and structural biotechnology journal*, 18:571–582, 2020.

[43] A. Saadatpour, I. Albert, and R. Albert. Attractor analysis of asynchronous boolean models of signal transduction networks. *Journal of theoretical biology*, 266(4):641–656, 2010.

[44] K. Klemm and S. Bornholdt. Stable and unstable attractors in boolean networks. *Physical Review E*, 72(5):055101, 2005.

[45] S. Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5(suppl_1):S85–S94, 2008.

[46] O. Brandman, J. E. Ferrell Jr, R. Li, and T. Meyer. Interlinked fast and slow positive feedback loops drive reliable cell decisions. *Science*, 310(5747):496–498, 2005.

[47] A. Garg, A. Di Cara, I. Xenarios, L. Mendoza, and G. De Micheli. Synchronous versus asynchronous modeling of gene regulatory networks. *Bioinformatics*, 24(17):1917–1925, 2008.

[48] R. Albert and R. Robeva. Signaling networks: Asynchronous boolean models. In *Algebraic and discrete mathematical methods for modern biology*, pages 65–91. Elsevier, 2015.

[49] R. Kang, D. Tang, N. E. Schapiro, K. M. Livesey, A. Farkas, P. Loughran, A. Bierhaus, M. T. Lotze, and H. J. Zeh. The receptor for advanced glycation end products (RAGE) sustains autophagy and limits apoptosis, promoting pancreatic tumor cell survival. *Cell Death & Differentiation*, 17(4):666–676, 2010.

[50] H. Gong, Q. Wang, P. Zuliani, J. R. Faeder, M. Lotze, and E. Clarke. Symbolic Model Checking of Signaling Pathways in Pancreatic Cancer. In *BICoB*, page 245, 2011.

[51] Z. Mai and H. Liu. Boolean network-based analysis of the apoptosis network: irreversible apoptosis and stable surviving. *Journal of theoretical biology*, 259(4):760–769, 2009.

[52] H.-D. Meyer, F. Gatti, and G. A. Worth. *Multidimensional quantum dynamics*. John Wiley & Sons, 2009.

[53] O. Koch and C. Lubich. Dynamical low-rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2):434–454, 2007.

[54] C. Lubich. *From quantum to classical molecular dynamics: reduced models and numerical analysis*. European Mathematical Society, 2008.

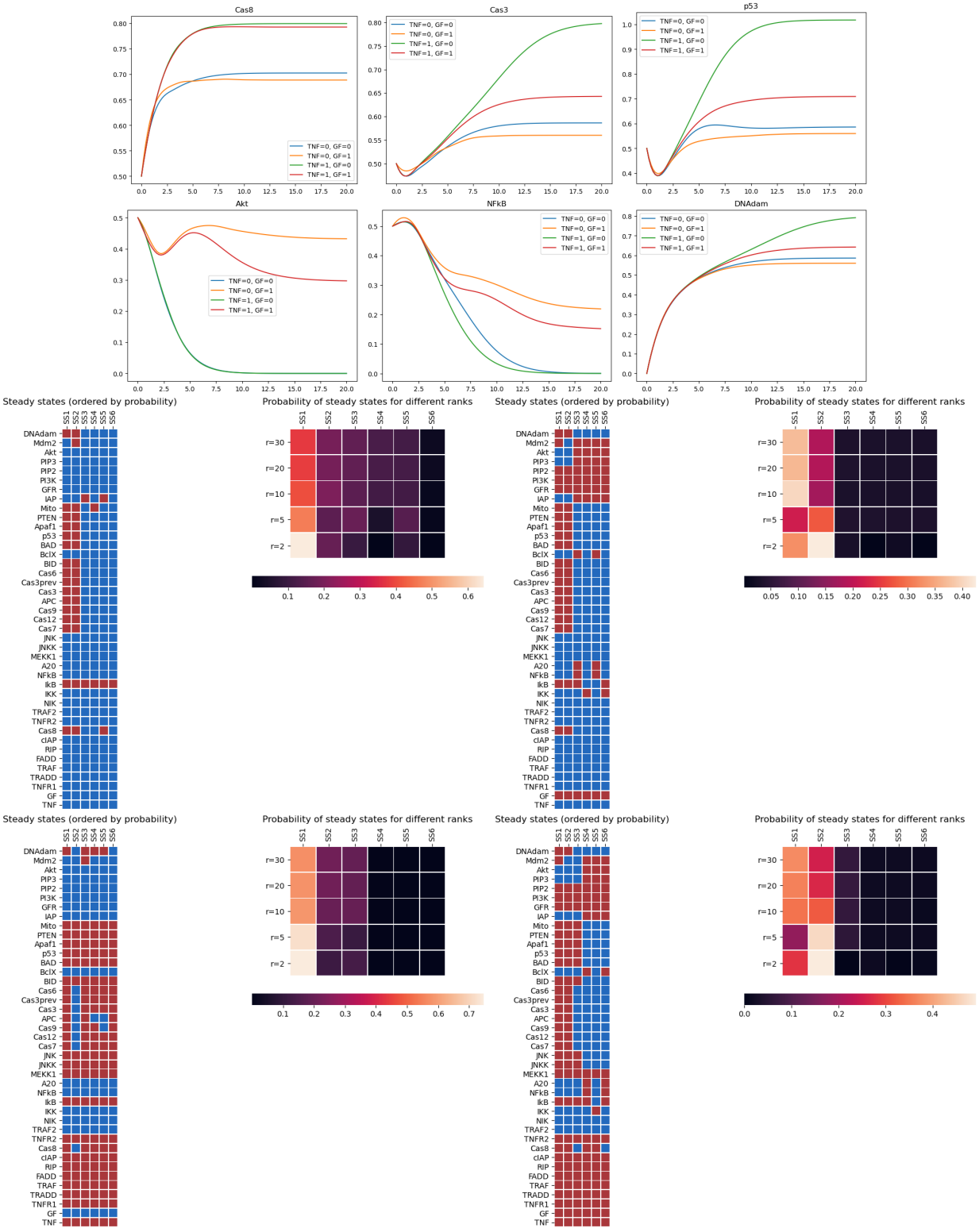


Figure 5: On the top the dynamics of selected species for the apoptosis model is shown for the four different configurations of *TNF* and *GF* (i.e. *TNF*, *GF* set to 00, 01, 10, and 11). On the bottom the steady states are depicted corresponding to the same four configurations (from top-left to bottom-right). The steady states are sorted according to their probability. Red represents that the species is on, blue represents that the species in this steady state is off.