On Quantitative Comparison of
Chemical Reaction Network Models

Ozan Kahramanoğulları

University of Trento, Department of Mathematics

`ozan-k.com`

Chemical reaction networks (CRNs) provide a convenient language for modelling a broad variety of biological systems. These models are commonly studied with respect to the time series they generate in deterministic or stochastic simulations. Their dynamic behaviours are then analysed, often by using deterministic methods based on differential equations with a focus on the steady states. Here, we propose a method for comparing CRNs with respect to their behaviour in stochastic simulations. Our method is based on using the flux graphs that are delivered by stochastic simulations as abstract representations of their dynamic behaviour. This allows us to compare the behaviour of any two CRNs for any time interval, and define a notion of equivalence on them that overlaps with graph isomorphism at the lowest level of representation. The similarity between the compared CRNs can be quantified in terms of their distance. The results can then be used to refine the models or to replace a larger model with a smaller one that produces the same behaviour or vice versa.

1   Introduction

Chemical reaction networks (CRNs) constitute an expressive language for building models that are dedicated to studying dynamical behaviour of biological systems. Many implementations of simulation languages for systems biology use CRN representations for their back-ends that perform deterministic or stochastic simulations, e.g., [2, 14, 12]. From a modelling, or programming, perspective, the main challenge in the context of the complex make-up of biological systems is to come up with simpler models that are effective in addressing biological queries. In such a setting, it is a common practise to compare different models that are built to produce a certain behaviour. The methods for comparison are often devised on a case-by-case basis, and they typically rely on the deterministic representations of the CRNs. The comparisons are commonly performed on the steady states. The notion and the extent of acceptable variation in behaviour between compared models depend on the system and the query in question.

Despite the limited number of formal means, drawing parallels between various models of biological systems is central to many investigations in this field. Given that existing efforts are often limited by the measurement of ad hoc model signals, it is inherently challenging to provide a general method for the task. To this end, we propose a methodology for comparing models in relation to the dynamic behaviour given by their flux graphs resulting from stochastic simulations [11, 8]. Here, flux is defined in terms of the intensity of resource flow during stochastic simulation from one reaction to another. The flux graphs are directed graphs that display how many of which network species instances flow between which model components during any chosen time interval of the simulation. As this provides a mathematical structure that quantifies the model dynamics, we use this information as a discrete abstraction of the model behaviour that can be compared with other structures obtained in the same way. The similarity between simulations with CRNs can then be quantified by resorting to a distance function, which we show to be a metric. In the following, we illustrate our method and discuss different models, whereby we argue that models separated with a smaller distance produce more similar behaviours.

2 Chemical reaction networks and mass action semantics

Let $\mathcal{N}$ denote a chemical reaction network (CRN) consisting of a set of N reactions defined on M number of species, and an initial state, which is a vector of the quantities of the M species. In $\mathcal{N}$, each reaction

$$m_1 R_1 + \ldots + m_l R_l \xrightarrow{\rho} n_1 P_1 + \ldots + n_r P_r$$
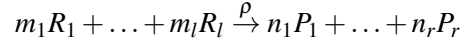
describes the reactants on the left-hand side that are replaced by the products on the right-hand side with rate ρ. The constants $m_1, \ldots, m_l$ and $n_1, \ldots, n_r$ are positive integers that denote the multiplicity of the reactants that are consumed and the products that are produced. When such a reaction occurs, it modifies the state vector. That is, if species $R_1, \ldots, R_l$ are present in their multiplicities $m_1, \ldots, m_l$, they are consumed, and the species $P_1, \ldots, P_r$ are produced, again in their respective multiplicities $n_1, \ldots, n_r$. The reaction rate constant ρ is a positive real number, which determines how often a reaction occurs in a system. According to the mass action kinetics, the probability of a reaction's firing at a particular state is proportional with ρ and the number of possible combinations of its reactants at that state.

Example 1 *The Lotka-Volterra network $\mathcal{N}_1$ models the interactions of a predator X with a prey Y.*

$$\{\, r_1 : X \xrightarrow{100.0} \cdot\,, \quad r_2 : Y \xrightarrow{300.0} 2Y\,, \quad r_3 : X + Y \xrightarrow{1.0} 2X \,\}$$
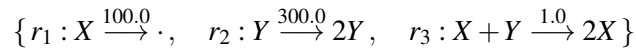
The network is initiated with 100 X and 100 Y individuals; the initial state is the vector $S_0 = \langle 100, 100 \rangle$.

Example 2 *The SIR network $\mathcal{N}_2$ models the spread of an infection in a population, whereby the individuals infect each other by direct interactions. An individual who recovers from the illness has immunity.*

$$\{\, r_1 : S + I \xrightarrow{\beta} I + I\,, \quad r_2 : I \xrightarrow{\gamma} R \,\}$$
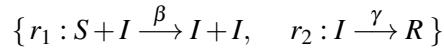
Each individual can be susceptible (S), infected (I), or resistant (R). Thus an initial state with 40 susceptible, 1 infected and no resistant is given by the vector $S_0 = \langle 40, 1, 0 \rangle$.

Stochastic trajectories of CRNs are commonly generated by using Gillespie's stochastic simulation algorithm (SSA) [6]. These simulations are then commonly depicted as time series. SSA is a Monte Carlo algorithm based on mass action with a continuous time Markov chain semantics. Thus, the simulations with SSA, at their limit, overlap with the deterministic ordinary differential equation simulations. However, stochastic simulations reflect the fluctuations in species numbers and extinctions that may arise due to low species numbers, which are not perceivable in the deterministic setting, e.g., as in Figure 1.
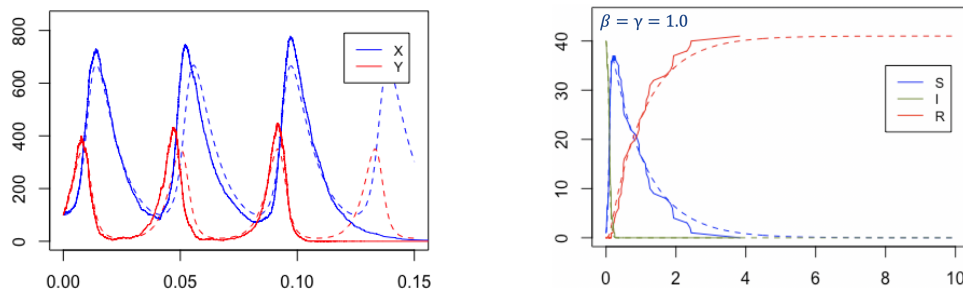

Figure 1: Time series of the CRNs in Examples 1 and 2. The dashed lines are deterministic simulations.
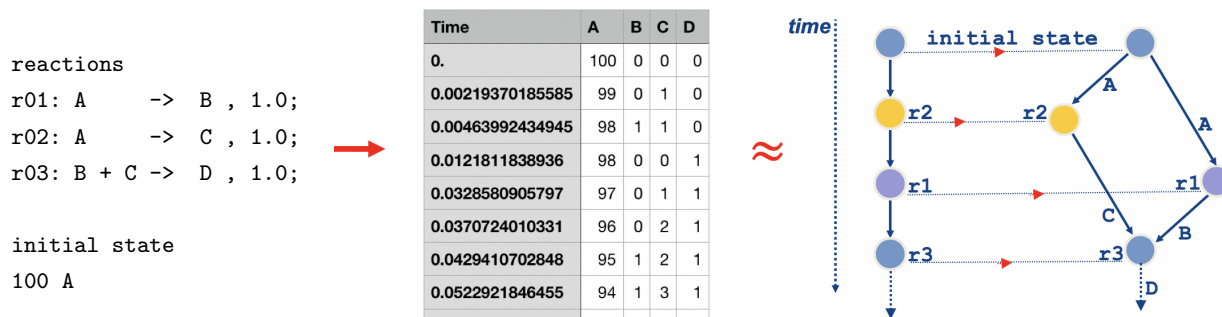
Figure 2: The fSSA algorithm [11, 8] extends the stochastic simulation algorithm [6] by logging the dependencies of each reaction instance by keeping track of the origin of each species that reaction instance consumes. This results in a directed acyclic graph that complements the time series data.

3   Flux graphs of chemical reaction networks

A stochastic simulation with a CRN can be considered as a mathematical reduction from a complex structure, i.e., the CRN, to a simpler structure, i.e., the simulation trajectory. As a result of this reduction, the algorithm delivers a sequential structure, whereby the reaction instances occur one after another starting from the initial state. Because of the Markov chain semantics, there cannot be any simultaneous reaction instances. Each reaction instance modifies the state at which it occurs and results in a new state. The time stamps of the reaction instances can be seen as a manifestation of the sequential total order structure. However, a different point of view can be obtained if we observe the reaction instances from the point of view of their interdependencies due to the resources that they consume and produce: each reaction instance consumes resources that were produced by another reaction, and produces others that become available for consumption. These dependencies result in a partial order structure or a directed acyclic graph (dag). When we superimpose this dag on to the time series, it becomes possible to treat the simulation trajectory as an interleaving of a concurrent computation, whereby the dag edges display the resource dependencies. A graphical visualisation of this idea is depicted in Figure 2 on a simple CRN.
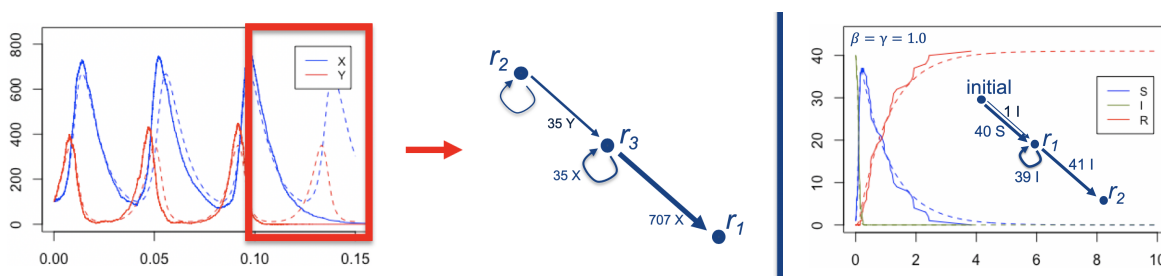

Figure 3: Flux graphs of the simulations in Figure 1. The graph on the left displays the fluxes between the time points 0.1 and 0.15, where the species go extinct. The graph on the right displays the fluxes of the network in Example 2 for the complete simulation.

In previous work, we have introduced a conservative extension of Gillespie's stochastic simulation algorithm [6], called fSSA, that generates these graphs. In addition to the time series, fSSA logs the dependency data by introducing a constant cost to each simulation step [11, 8]. The algorithm then folds these dags into *flux graphs* that provide a quantification of the flow of resources between reaction instances for any user-specified time interval. For the details of the algorithm we refer to [11, 8]. In the following, we describe the structure of the flux graphs, and how they are used for comparing CRNs.

The flux graphs are edge-coloured directed graphs, denoted by $\mathscr{F}[t,t']$, where t,t' denote the time points for the beginning and the end of the flux interval. Each edge of the graph is of the form $p \xrightarrow{x,n} q$, where p and q are nodes representing the reactions of the CRN, x is a network species, and n is the weight. The edge colour x,n on the arrow denotes that between time points t and t', species x flowed from p to q with a weight of n. The weight n denotes the multiplicity of the species x that are logged to have flowed from p to q. Figure 3 displays example flux graphs of the simulations in Figure 1.

4 Comparing chemical reaction networks

The method for comparing CRNs uses the flux graphs as discrete representations of the dynamic behaviours of the CRNs. Given two networks to be compared, the method takes two parameters.

The first parameter is the number of simulations, k, to be considered for the comparison. This parameter is used to run k number of simulations, and the mean fluxes of the k simulations is considered for the comparison. A smaller k exposes the stochastic effects and the noise in the system. In contrast, a larger k can be used to factor for the stochastic effects. This way, it becomes possible to expose the mean behaviour, similar to the use of deterministic simulations with ordinary differential equations. It is also possible to use different k parameters for the two compared networks.

The second parameter of the method is the time interval to be considered for the comparison, which can be the complete simulation duration or any other interval within it. Again, it is possible to use two different intervals for the two compared networks.

The method is performed by first running k simulations on the two CRNs, and computing the flux graphs of these simulations for the given intervals. The comparison is performed on the mean flux graphs of each set of simulations. This results in the two flux graphs, one for each network. Each flux graph is then normalised by the maximum flux in that network. This way, each weight n of each flux edge
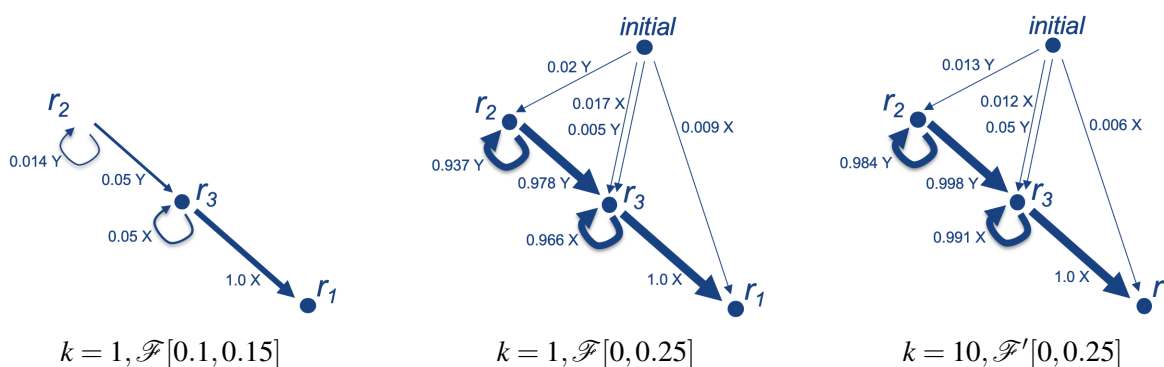

Figure 4: A comparison of the CRN in Example 1 with different time intervals and k parameters.

is replaced with a weight n/m, where m denotes the maximum flux value in that network. Thus, each weight in the flux graph takes a value between 0 and 1.0.

Example 3 *Let us consider the CRN in Example 1. The single run of a simulation depicted in Figure 1 results in an extinction, which is different from the deterministic behaviour of the system. A comparison of the normalised flux behaviour of the complete simulation with the extinction window in the time interval between 0.1 and 0.15 in Figure 3 reflects this contrast. As depicted in Figure 4, this contrast becomes even more amplified when we observe the mean flux behaviour of 10 simulations, which is closer to the deterministic setting. In particular, the normalised mean fluxes display a more balanced distribution between reactions in comparison to the fluxes in a single simulation that results in an extinction.*

Example 4 (sensitivity analysis) *The SIR network $\mathcal{N}_2$ has the rate parameters γ and β that are instantiated as 1.0 in the stochastic simulation depicted in Figure 1. The flux graph in Figure 3 together with the time-series plot shows that all the susceptible individuals become infected and they recover before the time point 4.0 is reached. To observe how the parameters γ and β influence the dynamic behaviour, we have performed a set of simulations by varying these two parameters within a spectrum of 4 orders of magnitude from 10^{-2} to 10^{2} in the logarithmic scale. This has resulted in $9 \times 9 = 81$ simulations. We have then visualised the effect of these variations on the two flux edges of this network for the interval from time 0 to 2, and compared these with the fluxes of the complete simulation. The process, whereby a previously infected individual infects another individual, is given by the flux edge $r_1 \xrightarrow{I} r_1$. The recovery of infected individuals is given by the fluxes to the reaction r_2, that is, the flux edge $r_1 \xrightarrow{I} r_2$. Figure 5*
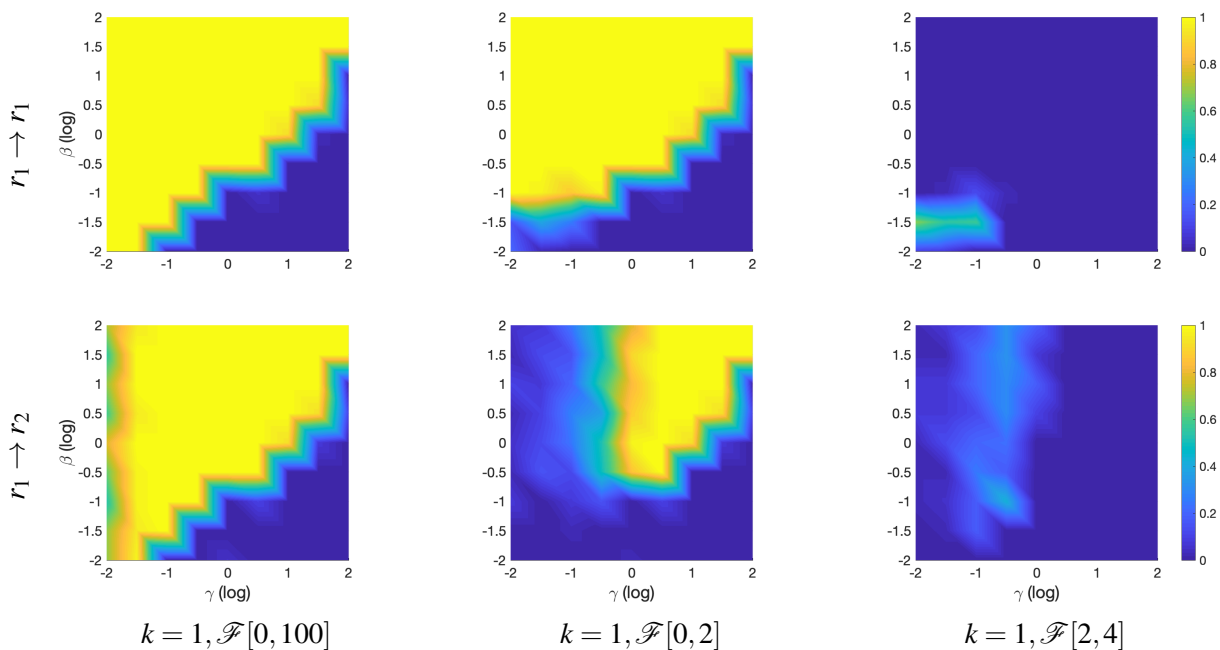

Figure 5: The mesh of the normalised fluxes resulting from 81 simulations in different time intervals, where the rates β and γ of $\mathcal{N}_1$ in Example 2 are scanned 4 orders of magnitude on the axes in logarithmic scale. The top row displays the $r_1 \xrightarrow{I} r_1$ fluxes and the bottom row displays the $r_1 \xrightarrow{I} r_2$ fluxes.

displays the normalised flux values in these simulations for the complete simulation as well as for the time intervals 0 to 2 and 2 to 4. As it can be seen in these plots, the fluxes in the shorter intervals ($\mathscr{F}[0,2]$ and $\mathscr{F}[2,4]$) add up to contribute to the fluxes of the complete simulation ($\mathscr{F}[0,100]$). While a speed up in α results in the infection of the whole population, a smaller γ delays the recovery.

The similarity between the two normalised flux graphs $\mathscr{F}$ and $\mathscr{F}'$ is determined by a distance function function. The distance between the two flux graphs $\mathscr{F}$ and $\mathscr{F}'$ is given by the sum of squared differences of the edge weights with each colour (species). If an edge with a colour does not exist, its weight is assigned a value of 0. The distance function δ is formally defined as follows:

$$\delta(\mathscr{F}_1, \mathscr{F}_2) = \mathsf{sqrt}\Big(\sum_{p \xrightarrow{x} q \,\in\, \mathscr{F}_1 \cup \mathscr{F}_2} (w_1 - w_2)^2 \ \text{ such that for } i = 1, 2, \ \ w_i = \begin{cases} n_i, & p \xrightarrow{x, n_i} q \in \mathscr{F}_i \\ 0, & otherwise \end{cases} \Big)$$

Proposition 5 *The distance function δ is a metric. That is, for any flux graph $\mathscr{F}_1$, $\mathscr{F}_2$ and $\mathscr{F}_3$:*

1. $\delta(\mathscr{F}_1, \mathscr{F}_2) \geq 0$;
2. $\delta(\mathscr{F}_1, \mathscr{F}_2) = 0$ *if and only if* $\mathscr{F}_1 = \mathscr{F}_2$;
3. $\delta(\mathscr{F}_1, \mathscr{F}_2) = \delta(\mathscr{F}_2, \mathscr{F}_1)$;
4. $\delta(\mathscr{F}_1, \mathscr{F}_3) \leq \delta(\mathscr{F}_1, \mathscr{F}_2) + \delta(\mathscr{F}_2, \mathscr{F}_3)$.

Proof: The function δ can be mapped to Euclidean distance by mapping flux graphs to Euclidean vectors as follows: for the flux graphs $\mathscr{F}_1$, $\mathscr{F}_2$ and $\mathscr{F}_3$, let $E = \{(p, x, q) \mid (p, (x, n), q) \in \mathscr{F}_1, \mathscr{F}_2, \mathscr{F}_3\}$ and $|E| = \ell$. For each $\mathscr{F}_i$, construct an ℓ dimensional vector by applying to each element of E the function

$$f(p, x, q) = \begin{cases} n & (p, (x, n), q) \in \mathscr{F}_i \\ 0 & otherwise. \end{cases}$$

□

Such a comparison of dynamic behaviour results in a spectrum of observations:

(*i*.) If the flux graphs are isomorphic, the differences in their edge weights will determine how similar they are, whereby identical flux graphs will have a distance of 0. We refer to isomorphism here, rather than simple equivalence, because the compared graphs might have different sets of vertices, for example, due to different naming.

(*ii*.) if the two flux graphs have overlapping edges, but they are not isomorphic, then their uncommon edges will increase their distance.

(*iii*.) If the flux graphs are completely different, every edge of the two flux graphs will contribute to their distance.

Below, we illustrate our method on networks with varying sizes and connectivities from the literature, whereby we argue that the networks separated with a smaller distance produce more similar behaviours.

5   Quantifying noise and comparing different time intervals of a CRN

The notion of distance between two flux graphs can be used to compare simulations of the same CRN to estimate the noise in the system. As an example for this let us consider the CRN depicted in Figure 6, which models the dynamic interactions of plasmin (`pls`) and urokinase-type plasminogen activator (`upa`)

[16]. This network does not have any species instances at the initial state as it contains the zero-order reactions r13, r14 and r15 that introduce into the system the urokinase (tcupa), plasminogen (plg) and the generic substrate species X. The metabolites pls and upa activate each other by interacting in their active and inactive forms. These interactions are modelled by the reactions r1, r2 and r3. The reactions r4, r5, r6, r7, r8 and r11 model the degradation and dilution of the metabolites. The complexation and catalysis of various substrates by pls are modelled by the reactions r8, r9 and r12.

Figure 6 displays the time series of two representative stochastic simulations with this network as well as their normalised flux graphs. For comparison, the stochastic time series are plotted together with the deterministic trajectory of the system. These two stochastic simulations demonstrate similar qualitative behaviours, however they differ in their quantitative behaviour due to the noise in the system. To quantify the difference that these fluctuations create, we have computed the distance of the normalised flux graphs in these two simulations. The resulting distance value of 0.0334 provides an estimate of the effect of noise that makes two such simulations different. We have then performed this procedure on 100 pairs of simulations, and obtained a mean distance of 0.0315 with a standard deviation of 0.01497.

By taking this estimated noise value as the base-line, we have compared the dynamic behaviour of the system in the transient interval at the first 100 time units with the steady state interval in the last 100


```
r01: scupa + plg ->
            pls + scupa   , 0.00006;
r02: pls + scupa ->
            tcupa + pls. , 0.0664;
r03: tcupa + plg ->
            pls + tcupa   , 0.0015;
r04: scupa    ->            , 0.084;
r05: plg      ->            , 0.032;
r06: pls      ->            , 0.084;
r07: tcupa    ->            , 0.084;
r08: pls + X -> Xpls      , 50.0;
r09: Xpls     -> X + pls  , 0.016;
r10: Xpls     -> pls.     , 0.02;
r11: X        ->            , 0.032;
r12: Xpls     ->            , 0.032;
r13:          -> scupa    , 1.9264;
r14:          -> plg      , 6.02;
r15:          -> X        , 6.02;
```
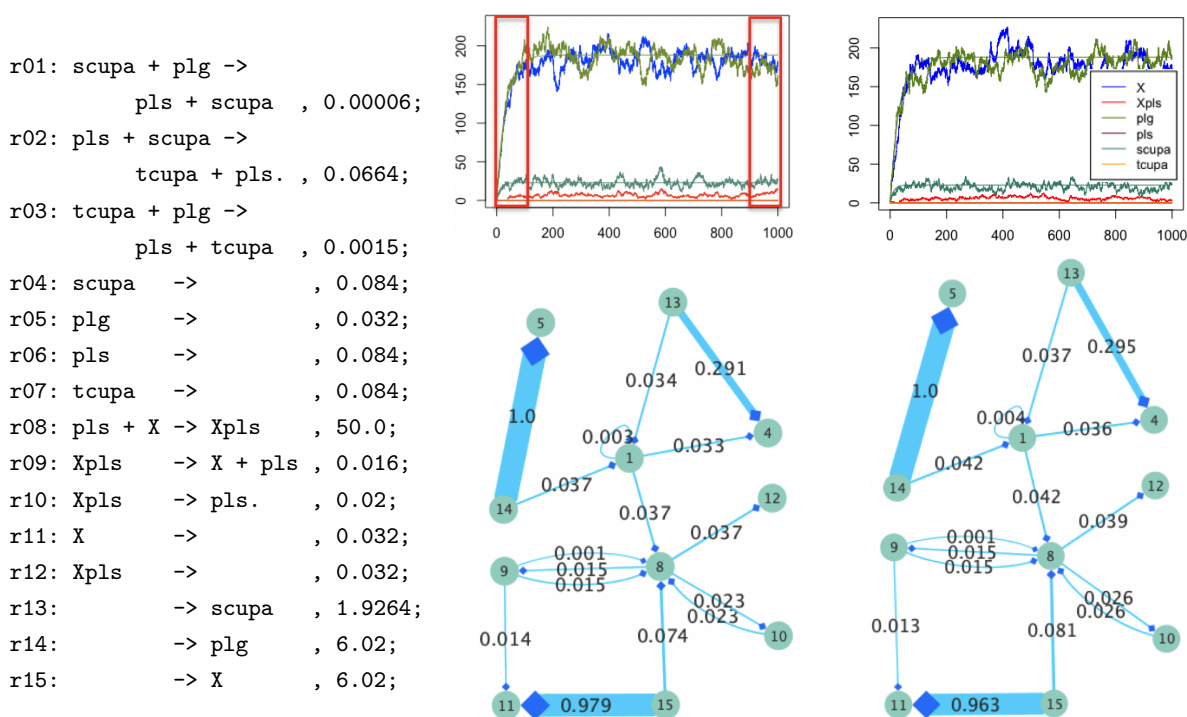
Figure 6: The CRN in [16] that models the interactions of plasmin (pls) and urokinase-type plasminogen activator (upa). The two representative plots display the differences in stochastic time-series trajectories. For comparison, the stochastic simulations are plotted together with the deterministic trajectory. The transient interval within the first 100 time units, whereby the system approaches the steady state, and the steady state interval between the time units 900 and 1000 are framed. The normalised flux graph of these two simulations are depicted below their time series. The flux graphs are rendered with Cytoscape [15].

time units, which can be observed in the deterministic trajectory. These two intervals are depicted in frames in the time series of the first simulation in Figure 6. Over a sample of 100 simulations the mean distance between these two intervals is 0.1396 with a standard deviation of 0.0398. This distance value is greater than the estimated noise by a factor of 4.43 on a sample of same size, and thus indicates the extent to which the dynamic behaviour at the transient interval differs from the equilibrium behaviour.

```
reactions
r01: A + R  -> RA     , 1.0;        r11: RDE -> E + RD , 0.136;      r21: R   -> RT , 50.0;
r02: A + RD -> RDA    , 1.0;        r12: RTE -> E + RT , 76.8;       r22: RT  -> R  , 0.02;
r03: A + RT -> RTA    , 1.0;        r13: RTA -> RDA    , 2104;       r23: RTE -> RDE , 0.02;
r04: RA     -> A + R  , 500;        r14: RDA -> RA     , 0.02;       r24: RDE -> RE  , 6.0;
r05: RDA    -> A + RD , 500;        r15: RA  -> RDA    , 5.0;        r25: RE  -> RDE , 1.65;
r06: RTA    -> A + RT , 3.0;        r16: RA  -> RTA    , 4.25;       r26: RE  -> RTE , 50;
r07: E + R  -> RE     , 0.43;       r17: RTA -> RA     , 0.0002;     r27: RTE -> RE  , 0.02;
r08: E + RD -> RDE    , 0.0054;     r18: RT  -> RD     , 0.02;
r09: E + RT -> RTE    , 0.0075;     r19: RD  -> R      , 0.02;       initial state
r10: RE     -> E + R  , 1.074;      r20: R   -> RD     , 1.65;       1000 R; 776 E; 10 A;
```
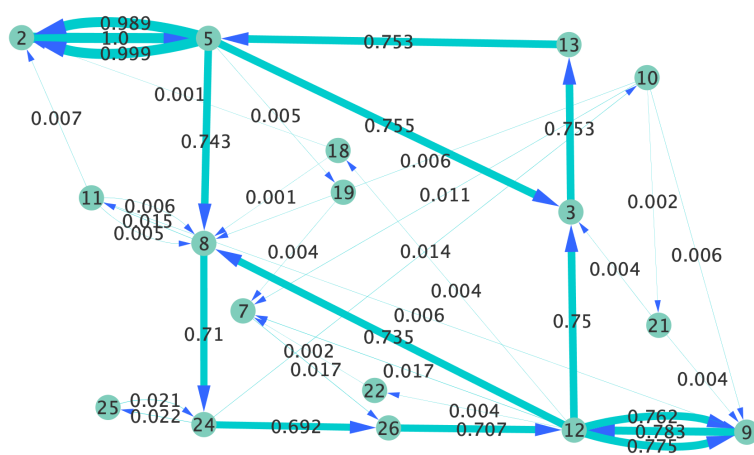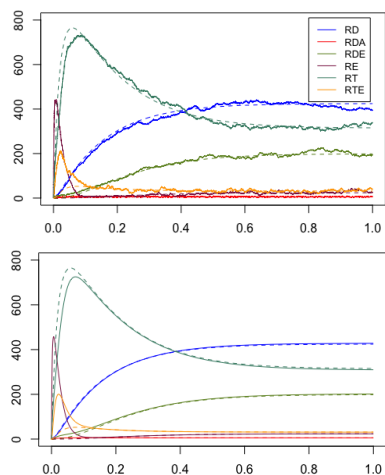

Figure 7: The CRN that models the Rho GTP-binding proteins and its flux graph [3, 11]. In the time series on the top, a stochastic simulation trajectory with this network is plotted together with the deterministic ordinary differential equations trajectory, which is indicated with the dashed lines. The reactions 2, 3, 5, 8, 9, 12, 13, 24, 26 that participate in the strong fluxes in the flux graph and the reaction 21 that introduces the active metabolite RT are used to construct a simplified network. The time series below displays the deterministic time series with the full network together with the deterministic time series of the reduced network, plotted with the dashed lines. The flux graph is rendered with Cytoscape [15].

6   Quantifying information loss in model simplification

In systems biology, it is often desirable to replace a model with a simpler one that produces a similar dynamics with the minimum loss of information. As an example for this, let us consider the CRN in Figure 7 that models the Rho GRP binding proteins that cycle between inactive (RD) and active (RT) forms by being catalysed by the enzymes A and E. In previous work [11], we have shown that flux analysis can be used to simplify this network to its subset that consists of the reactions participating in strong flux edges. Such a simplified network, consisting of the reactions 2, 3, 5, 8, 9, 12, 13, 21, 24 and 26, produces a similar time series as shown in Figure 7. Here, although reaction 21 does not participate in a strong flux, it is included in the simplified network as it introduces the active metabolite RT.

To quantify the loss of information in the simplification, we have first computed the distance between the simulations of the full network to obtain an estimate of noise. On a sample of 100 pairs of simulations, the mean distance between two simulations of the full network is 0.1135 with a standard deviation of 0.05723. The distance between the complete network and the simplified network for the full interval, on the other hand, accounts for a distance of 1.13 on a sample of 100 simulations. When we compare the transient interval up to the time point 0.5, this distance becomes 1.74. However, the distance in the steady state interval between the time points 0.5 and 1.0, again on a sample of 100 simulations, is 0.0987, which is in the order of noise base-line.

These results demonstrate that for the steady state interval the simplified network's information loss is in the order of noise base-line. Moreover, the complete loss of information in the simplified network is due to the transient interval, whereby the system species stabilise while approaching equilibrium.

7   Quantifying variation in behaviour in different regimes of a network

A CRN with a certain set of reactions can have drastically different time series behaviours with variations in initial conditions or rate constants. Because such parameters are used to represent metabolic conditions in different regimes, quantifying the deviation in behaviour that they cause can provide insights for the modelled biological systems. An application of this idea is illustrated in Example 4 and Figure 5.

Let us now consider a larger CRN, depicted in Figures 8 and 9, that models the Gemcitabine biochemical machinery [10]. Gemcitabine is a prodrug, which is widely used for treating various carcinomas. Gemcitabine (dFdC) exerts its clinical effect by incorporating its triphosphate metabolite (dFdC-TP) into DNA, thereby inhibiting DNA synthesis and replication, eventually resulting in programmed cell death. This process takes place in competition with the natural nucleoside dCTP. Besides the indirect competition for incorporation into DNA, dFdC-TP and dCTP compete by direct competition, whereby they inhibit the enzymes in each other's activation cascades. The relative abundance of these enzymes, RR and dCK, gives rise to an efficacy/non-efficacy spectrum of the drug on a continuum. In the CRN in Figure 9, this continuum is modelled by changes in the rate constants α and β of the reactions 25 and 27. On one extreme, the efficacy regime is given by $\alpha = 0.000001$ and $\beta = 0.01$. On the other extreme, the non-efficacy regime is given by $\alpha = 10$ and $\beta = 0.000001$. These two regimes result in drastically different time series for the metabolites dFdC_TP_DNA and dCTP_DNA that determine the efficacy of the drug.

To estimate the behavioural difference between the efficacy and non-efficacy regimes, we have first computed the mean distance of the normalised flux graphs within these regimes as the base-line. On a sample of 100 pairs of simulations, the mean distance between two simulations in the efficacy regime is 0.0082 with a standard deviation of 0.0035. In the non-efficacy regime, the distance is 0.1364 with a standard deviation of 0.081. On this sample of 100 simulations, the mean distance between the efficacy

and non-efficacy regimes is 1.48. When we take the efficacy regime as the base-line for comparing the efficacy and non-efficacy regimes, we obtain the ratio 1.48/0.1364 that delivers a factor of 10.85. Taking the non-efficacy regime as the base-line results in the ratio of 1.48/0.0082, delivering a factor of 180.48. This allows us to use this later factor as a quantification of the upper estimate for the variation between the two regimes at the extremities of the efficacy/non-efficacy spectrum.

8 Discussion

CRNs provide a convenient and expressive programming language for modelling biological systems as well as other complex systems across many branches of science. Besides their role as back-end engines in rule-based modelling languages [2, 14], their use in synthetic biology in implementing molecular programs resembles common programming practises [12, 9]. However, the quantitative nature of CRNs, which heavily relies on parameterisation, makes them difficult to analyse and compare in a qualitative manner. This is because, unlike programs in common programming languages, the complex behaviours that emerge from CRN runs, i.e., simulations, often remain undetectable from their structure.

Despite the inherent challenges, the methods that originate from computer science, in particular those in concurrency theory, have been studied by various authors with the aim of connecting network structure with quantitative behaviour. These efforts mainly rely on the consideration that the analogy between complex systems studied in computer science and those in other disciplines is prone to providing new insights that can benefit both sides. In this respect, the flux graphs of our method has its origins in concurrency theory due to the resemblance between event structures [13, 7] and Markov chain semantics of CRNs: the dependency graphs (Figure 2), which are precursors of the flux graphs, can be treated as configurations in event structures. Such a view of these structures as well as those from other branches of computer science can provide further perspectives. In this regard, an approach that also uses partial order representations of traces for comparing concurrent programs can be found in [1]. Similar considerations that use distance metrics on various aspects of programs can provide insights in comparing programs in a quantitative manner for the cases, where a qualitative comparison is not straight-forward or meaningful.

In quantitative disciplines, being able compare the primary structures of function, possibly in a way that permits for a notion of equivalence, is a key to evaluate and gain insight on such structures. Our method for comparing CRNs with respect to their dynamic behaviours in simulations is set out with such an ambition. In the context of comparing CRNs, a promising method that is orthogonal to the one proposed here and combines quantitative and structural points of views is given in [5], where the authors describe a method, with certain constraints, for minimising a CRN by aggregating its species. The notion of species aggregation as well as reaction aggregation is a topic of ongoing investigation also for the present work. In particular, such aggregations can provide further aid when comparing very large networks or comparing the behaviour of a large network with a smaller one. Another related topic, which is left for future work, concerns algorithms for re-mapping reaction nodes of compared flux graphs. This is because two networks that appear different at a first observation can be separated by a small distance after renaming their reaction edges, and this can explain the similarity in their dynamic behaviours.

Our method uses flux graphs resulting from stochastic simulations as discrete abstract representations of dynamic behaviour. This makes it possible to quantify the simulations in terms of their distance at steady states as well as at transient time intervals at which the system species have not yet reached equilibrium. The use of a metric in our method makes it possible to assign a distance measure to the comparisons that overlaps with graph isomorphism at the closest distance. Because flux graphs have intuitive interpretations as they display which portion of simulation species flows between which reactions,

they constitute favourable abstractions of behaviour that preserve the inherent stochasticity of the CRNs. Our comparison of flux graphs uses the notion that graphs that are close according to our metric have similar intensities of fluxes. A deeper analysis on when and why two graphs are close according to the metric is left for future work.

Due to the underlying mass action semantics, the observations obtained with our method complement those that are obtained with deterministic simulations with ordinary differential equations. Although differential equations do not directly capture noise and stochasticity, in some cases methods based on linear noise approximation [4] can be used for recovering the information on stochasticity. Still, the discrete structure of flux graphs makes them favourable structures to work with algorithmically [1], and their counter-part in the continuous setting of ordinary differential equations is yet to be identified.

References

[1] A. Bouajjani, C. Enea & S. Lahiri (2017): *Abstract Semantic Diffing of Evolving Concurrent Programs*. In: *International Static Analysis Symposium SAS 2017: Static Analysis*, LNCS 10422, pp. 46–65, doi:10.1007/978-3-319-66706-5_3.

[2] P. Boutillier, J. Feret, J. Krivine & W. Fontana (2018): *The Kappa Language and Kappa Tools, A User Manual and Guide,v4*. Available at https://kappalanguage.org/sites/kappalanguage.org/files/inline-files/Kappa_Manual.pdf.

[3] L. Cardelli, E. Caron, P. Gardner, O. Kahramanoğulları & A. Phillips (2009): *A Process Model of Rho GTP-binding Proteins*. Theoretical Computer Science 410/33-34, pp. 3166–3185, doi:10.1016/j.tcs.2009.04.029.

[4] L. Cardelli, M. Kwiatkowska & L. Laurenti (2016): *Stochastic Analysis of Chemical Reaction Networks Using Linear Noise Approximation*. Biosystems 149, pp. 26–33, doi:10.1016/j.biosystems.2016.09.004.

[5] L. Cardelli, M. Tribastone, M. Tschaikowski & A. Vandin (2017): *Maximal aggregation of polynomial dynamical systems*. Proceedings of the National Academy of Sciences, doi:10.1073/pnas.1702697114.

[6] D. T. Gillespie (1977): *Exact Stochastic Simulation of Coupled Chemical Reactions*. The Journal of Physical Chemistry 81(25), pp. 2340–2361, doi:10.1021/j100540a008.

[7] O. Kahramanoğulları (2009): *On Linear Logic Planning and Concurrency*. Information and Computation 207, pp. 1229–1258, doi:10.1016/j.ic.2009.02.008.

[8] O. Kahramanoğulları (2017): *Quantifying Information Flow in Chemical Reaction Networks*. In: *Proceedings of 4th International Conference on Algorithms for Computational Biology, AlCoB 2017*, LNCS 10252, pp. 155–166, doi:10.1007/978-3-319-58163-7_11.

[9] O. Kahramanoğulları & L. Cardelli (2015): *Gener: a minimal programming module for chemical controllers based on DNA strand displacement*. Bioinformatics 31 (17), pp. 2906–2908, doi:10.1093/bioinformatics/btv286.

[10] O. Kahramanoğulları, G. Fantaccini, P. Lecca, D. Morpurgo & C. Priami (2012): *Algorithmic modeling quantifies the complementary contribution of metabolic inhibitions to gemcitabine efficacy*. PLoS ONE 7(12), p. e50176, doi:10.1371/journal.pone.0050176.

[11] O. Kahramanoğulları & J. Lynch (2013): *Stochastic Flux Analysis of Chemical Reaction Networks*. BMC Systems Biology 7(133), doi:10.1186/1752-0509-7-133.

[12] M. R. Lakin, S. Youssef, F. Polo, S. Emmott & A. Phillips (2011): *Visual DSD: a design and analysis tool for DNA strand displacement systems*. Bioinformatics 27, pp. 3211–3213, doi:10.1093/bioinformatics/btr543.

[13] M. Nielsen, G. Plotkin & G. Winskel (1981): *Petri Nets, Event Structures and Domains, part 1*. Theoretical Computer Science 13(1), pp. 85–108, doi:10.1016/0304-3975(81)90112-2.

---

[1]The python scripts and tools of our method are available for download at ozan-k.com/distance.zip.

[14] J. A. P. Sekar & J. R. Faeder (2012): *Rule-Based Modeling of Signal Transduction: A Primer*. Computational Modeling of Signaling Networks. Methods in Molecular Biology (Methods and Protocols) 880, pp. 139–218, doi:10.1007/978-1-61779-833-7_9.

[15] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski & T. Ideker (2003): *Cytoscape: a software environment for integrated models of biomolecular interaction networks*. Genome Research 13(11), pp. 2498–2504, doi:10.1101/gr.1239303.

[16] L. Venkatraman, H. Li, C. F. Jr. Dewey, J. K. White, S. S. Bhowmick, H. Yu & L. Tucker-Kellogg (2011): *Steady states and dynamics of urokinase-mediated plasmin activation in silico and in vitro*. Biophysical Journal 101 (8), pp. 1825–1834, doi:10.1016/j.bpj.2011.08.054.
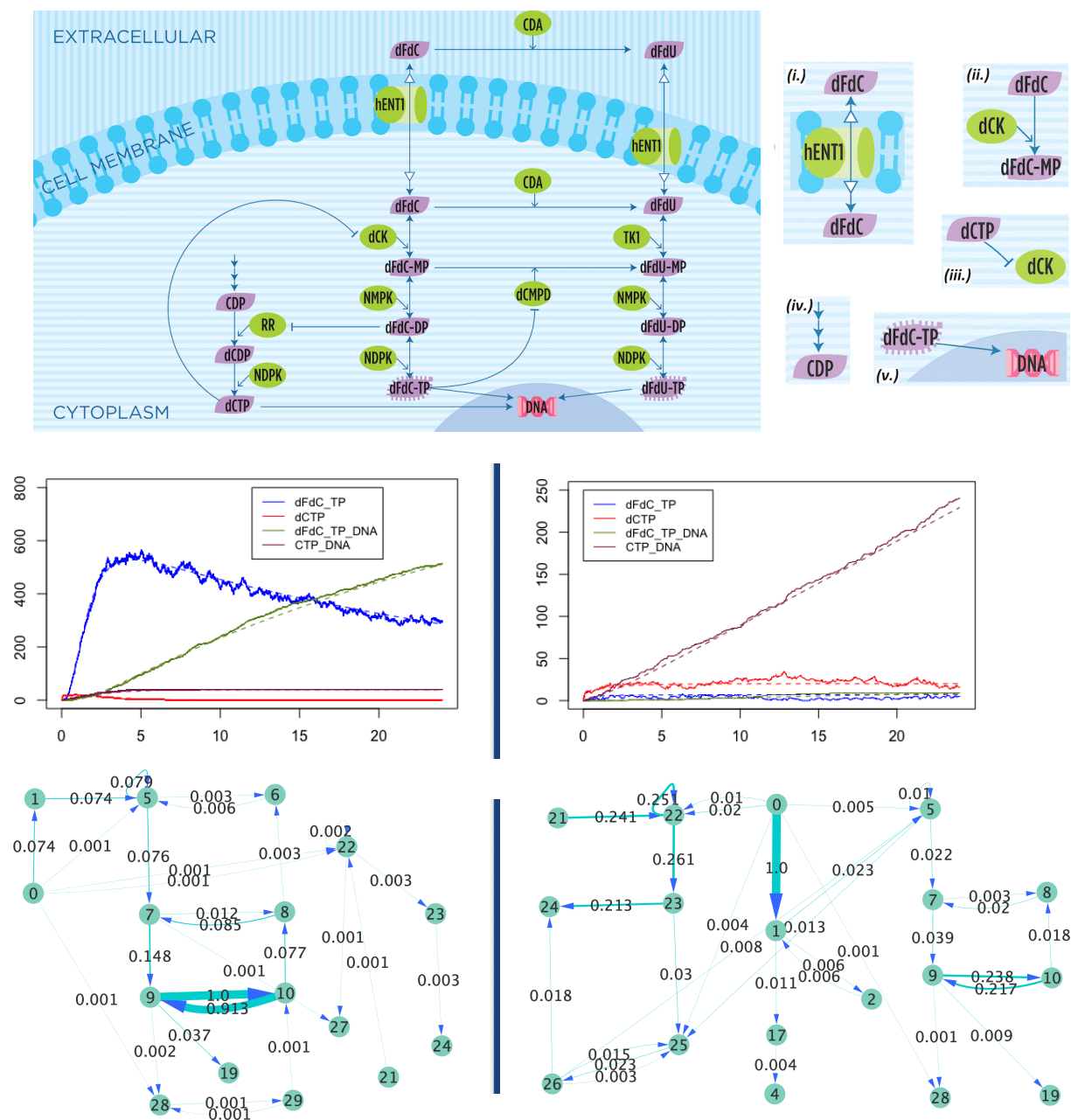
Figure 8: The molecular machinery of the Gemcitabine metabolism, representative time series plots in the extremities of the efficacy/non-efficacy continuum, and the mean normalised flux graphs on a sample of 100 simulations in these regimes [10]. The arrows on the right describe (*i.*) the transport into the cell, (*ii.*) enzymatic reactions, (*iii.*) inhibition, (*iv.*) synthesis, and (*v.*) DNA incorporation. The corresponding CRN is shown in Figure 9. The time series plot on the left for the efficacy regime is obtained from a single simulation with $\alpha = 0.000001$ and $\beta = 0.01$. The one on the right for the non-efficacy regime is obtained from a single simulation with $\alpha = 10$ and $\beta = 0.000001$. For comparison, both plots include the deterministic ordinary equation trajectories indicated with dashed lines. The flux graphs of the efficacy regime on the left and non-efficacy regimes on the right are rendered with Cytoscape [15].

```
reactions
r01: dFdC_out            -> dFdC                  , 9.97234;
r02: dFdC                -> dFdC_out              , 0.000261675;
r03: dFdC_out            -> dFdU                  , 0.00000472336;
r04: dFdU                -> dFdU_out              , 0.0508194;
r05: dFdC + dCK          -> dFdC_MP + dCK         , 0.104994;
r06: dFdC_MP             -> dFdC                  , 0.0875208;
r07: dFdC_MP             -> dFdC_DP               , 2.37162;
r08: dFdC_DP             -> dFdC_MP               , 0.212216;
r09: dFdC_DP             -> dFdC_TP               , 2.52037;
r10: dFdC_TP             -> dFdC_DP               , 1.44908;
r11: dFdU + dCK          -> dFdU_MP + dCK         , 0.0968;
r12: dFdU_MP             -> dFdU                  , 0.0000560415;
r13: dFdU_MP             -> dFdU_DP               , 0.07844;
r14: dFdU_DP             -> dFdU_MP               , 0.00420541;
r15: dFdU_DP             -> dFdU_TP               , 0.164322;
r16: dFdU_TP             -> dFdU_DP               , 0.0000905139;
r17: dFdC                -> dFdU                  , 0.000476746;
r18: dFdC_MP + dCMPD     -> dFdU_MP + dCMPD       , 0.0004559;
r19: dFdC_TP             -> dFdC_TP_DNA           , 0.0544456;
r20: dFdU_TP             -> dFdU_TP_DNA           , 0.000737496;
r21:                     -> CDP                   , 10;
r22: CDP + RR            -> dCDP + RR             , 5;
r23: dCDP                -> dCTP                  , 25.2037;
r24: dCTP                -> CTP_DNA               , 0.5;
r25: dCTP + dCK          -> dCTP_dCK              , alpha;
// efficacy: alpha = 0.000001 ~ non-efficacy: alpha = 10

r26: dCTP_dCK            -> dCTP + dCK            , 0.1;
r27: dFdC_DP + RR        -> dFdC_DP_RR            , beta;
// efficacy: beta = 0.01 ~ non-efficacy: beta = 0.000001

r28: dFdC_TP + dCMPD     -> dFdC_TP_dCMPD         , 0.001;
r29: dFdC_TP_dCMPD       -> dFdC_TP + dCMPD       , 0.1;

initial state
1000 dFdC_out; 10 dCK;  10 dCMPD;   20 CDP;   10 RR;
```

Figure 9: The CRN that implements the model depicted in Figure 8.