Neuro-Symbolic Discovery of Markov Population Processes

Luca Bortolussi LBORTOLUSSI@UNITS.IT
Department of Mathematics, Informatics and Geoscience, University of Trieste, Italy

Francesca Cairoli FRANCESCA.CAIROLI@UNITS.IT
Department of Mathematics, Informatics and Geoscience, University of Trieste, Italy

Julia Klein JULIA.KLEIN@UNI-KONSTANZ.DE
Department of Computer and Information Science, University of Konstanz, Germany
Centre for the Advanced Study of Collective Behaviour, University of Konstanz, Germany

Tatjana Petrov TATJANA.PETROV@UNITS.IT
Department of Mathematics, Informatics and Geoscience, University of Trieste, Italy
Max Planck Institute of Animal Behavior, Radolfzell, Germany
Centre for the Advanced Study of Collective Behaviour, University of Konstanz, Germany

Editors: G. Pappas, P. Ravikumar, S. A. Seshia

Abstract

Markov population processes (MPPs) are the natural modeling choice in various application domains where multiple interacting entities evolve stochastically over time, including biology, queueing theory, finance, and robotics. Motivated by real-world scenarios where time-series data for MPP models is increasingly available, we here employ a neuro-symbolic approach for discovering explanations of such data in terms of local, agent-to-agent interactions. Concretely, we assume that equidistant time-series measurements of a Markov population chain are given. Then, we propose how to automatically learn the explanatory models written in form of Chemical Reaction Networks (CRNs). Our approach is to use a symbolic representation of a CRN in form of a weighted bipartite graph, and to employ a graph-based Variational Autoencoder (VAE) to jointly infer both the interactions and the accompanying kinetic parameters. We demonstrate our proposed framework over three simple case studies. Our contribution represents a proof-of-concept that interpretable models of complex dynamical systems can be discovered in a fully automated and data-driven fashion, and it is applicable both in scenarios where data is available via experiments, or when it is generated by a black-box simulator.

Keywords: Chemical Reaction Networks; Variational Autoencoders; Graph Neural Networks.

1. Introduction

Markov population processes (MPPs) are the natural modeling choice in various application domains where multiple interacting entities evolve stochastically over time, including biology, queueing theory, finance, robotics. With the growing availability of data, advances in measurement technology, and the rise of deep neural network-based black-box emulators for complex systems, automating the discovery of explanations for population-level trajectories is becoming timely and compelling. In particular, understanding how agents interact locally is crucial not only for model interpretation but also for system design, verification, and control, thereby finding relevance in emerging areas such as explainable AI (Rudin, 2019) or precision medicine (Martinelli et al., 2021).

In this work, we assume that equidistant time-series measurements of a Markov population chain are given. Then, we propose a method to automatically learn the interaction mechanisms

written in form of Chemical Reaction Networks (CRNs). CRNs are a simple yet versatile formalism able to simulate any computable function (Cook et al., 2009). A typical example of a CRN is an epidemiological model SIR: the susceptible agent (S) gets infected whenever it interacts with another infected agent ($S + I \to 2I$), and an infected agent can spontaneously recover ($I \to R$). Each reaction is equipped with a constant kinetic rate parameter that specifies the relative speed of reactions competing in a well-stirred (spatially homogeneous) environment. The population dynamics of a CRN model evolves stochastically as a continuous-time Markovian process (Gillespie, 1977). Typically, the reactions of a CRN are proposed by the domain expert, and the kinetic rates are estimated from experimental data. Yet, even the simplest CRNs easily become computationally challenging, and a large body of work has dealt with abstractions allowing for scalable executions of CRNs, verification (model-checking), control of CRNs, or inferring kinetic rates from data (Öcal et al., 2019). Notably, recent efforts leverage machine learning approaches (Wen et al., 2023). Still, all these approaches are typically black-box abstractions that do not inform on the local mechanisms underlying the trajectories. In this work, our goal is to automatically infer an explicit representation of a CRN, that is, to infer the reactions and the accompanying kinetic rates, in a purely data-driven manner. Our approach is to use a symbolic representation of a CRN in form of a bipartite graph, and to employ a graph-based Variational Autoencoder (VAE) (Scarselli et al., 2009) to jointly infer the interactions and the accompanying kinetic parameters. The encoder infers the CRN reactions and the decoder infers the reaction rates, and thereby the dynamics. We use the linear noise approximation algorithm (Singh and Grima, 2017) to compute the first two moments of the stochastic dynamics. Our contribution represents a proof-of-concept that the microscopic interactions between agents can be inferred from macroscopic, population-level data in a fully automated and data-driven fashion. It is applicable both to scenarios where data is available via experiments, or when it is generated by a black-box system.

Related works. CRNs have long been studied across various disciplines, most notably in the context of systems and synthetic biology, epidemiology, and swarm robotics. Our idea for the graph-based encoding is inspired by Neural Relational Inference (NRI) (Kipf et al., 2018). Related works on CRNs include black-box emulations of trajectories induced by modeling gene regulation and signal transduction in cells (Bortolussi and Palmieri, 2018; Bortolussi and Cairoli, 2019; Repin and Petrov, 2021; Gupta et al., 2021; Cairoli et al., 2023; Cao et al., 2024). In (Martinelli et al., 2021), the authors propose learning deterministic CRNs from time-series data via statistical inference. Synthesis of CRNs from high-level formal specification properties was proposed in (Cardelli et al., 2017). To the best of our knowledge, we here propose the first technique for purely data-driven inference of CRNs from time-series data. Several related techniques have been proposed for temporal causal structure and representation learning (Barnard and Page, 2018; Li et al., 2024; Bramley et al., 2018; Chen et al., 2024; Rahmani and Frossard, 2023; Heinze-Deml et al., 2018; Valentin et al., 2020; Ng et al., 2019; Rottman and Keil, 2012; Absar and Zhang, 2021; Liu et al., 2010), differing in their semantic interpretation and underlying theoretical foundations.

2. Background

2.1. Chemical Reaction Network

Consider a Chemical Reaction Network (CRN) with N *species*, denoted as $S_1, \dots, S_N$, that interact according to M *reactions*, denoted as $R_1, \dots, R_M$, and let $\mathbf{x}(t) = (x_1(t), \dots, x_N(t)) \in \mathcal{X} \subseteq \mathbb{N}^N$

denote the state of the population at time t, i.e. the count of individuals (or entities) of each species present in the population at time t. Borrowing the chemical formalism, a reaction R_j, with $j \in \{1, \dots, M\}$, is symbolically described as

$$R_j : \quad a_{1j}S_1 + \cdots + a_{Nj}S_N \xrightarrow{\alpha_j} b_{1j}S_1 + \cdots + b_{Nj}S_N, \tag{1}$$

where a_{ij} and b_{ij} are known as *stoichiometric coefficients*, i.e., the number of entities of a species that are consumed or produced in a reaction. The *propensity function* $f_j(\mathbf{x})$ determines the rate of reaction R_j when the system is in state $\mathbf{x}$ and includes *rate parameter* α_j. The *update vector* ν_j describes the change of the system's state when reaction R_j fires and it is defined as $\nu_j := [b_{1j} - a_{1j}, \dots b_{Nj} - a_{Nj}]$. We denote the volume of the system by Ω, meaning the total number of individuals in the population. The dynamics of the CRN are described by the Chemical Master Equation (CME), i.e., the probability of finding the system in state $\mathbf{x}$ at time t given that it was in state $\mathbf{x}_0$ at time t_0 can be expressed as:

$$\frac{\partial p_{\mathbf{x}_0}\big(\mathbf{x}(t) = \mathbf{x}\big)}{\partial t} = \sum_{j=1}^{M} \Big[f_j(\mathbf{x} - \nu_j) p_{\mathbf{x}_0}\big(\mathbf{x}(t) = \mathbf{x} - \nu_j\big) - f_j(\mathbf{x}) p_{\mathbf{x}_0}\big(\mathbf{x}(t) = \mathbf{x}\big) \Big], \tag{2}$$

for every $\mathbf{x}_0, \mathbf{x} \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{N}^n$ denotes the state space and α_j denotes the parameter for the propensity function $f_j(\mathbf{x})$. The CME describes the flow of probabilities. Therefore, the underlying CRN dynamics are inherently stochastic. In a fixed volume, under the well-stirred assumption and the continuum hypothesis, a simple description of reaction rates is provided by the law of mass action: the reaction rate is proportional to the product of the concentrations of the reactants. The constant of proportionality is called the rate constant. Under the *mass action law*, the propensity functions can be written as $f_j(\mathbf{x}) = \alpha_j \prod_{i=1}^{N} \binom{x_i(t)}{a_{ij}}$. In general, the CME is a system with countably many differential equations, therefore, its analytic or numeric solution is almost always unfeasible. Since the CME cannot be computed exactly, we use linear noise approximation (LNA) to approximate the probability flow of the CRN's dynamics with a Gaussian distribution (Singh and Grima, 2017), where the first two moments can be numerically integrated.

Linear Noise Approximation. The LNA builds on the ansatz that fluctuations around the average of the counting process of an MPP are of order $\Omega^{1/2}$, where Ω is the population size (a.k.a. the volume). The underlying idea is that, in the macroscopic limit (limit of large individual numbers), we can say that (by Central Limit Theorem and the Law of Large Numbers) the standard deviation of noise roughly scales as the square root of the number of individuals: $\hat{\mathbf{x}}^{(\Omega)}(t) \approx \mu(t) + \Omega^{-1/2}\xi(t)$, where $\hat{\mathbf{x}}^{(\Omega)}(t)$ denotes the normalized state, i.e., $\hat{\mathbf{x}}^{(\Omega)}(t) = \frac{\mathbf{x}(t)}{\Omega}$, and $\xi(t)$ is a noise term. The *fluid approximation* of the CME is then expressed by the following ODE:

$$\frac{d\mu(t)}{dt} = F(\mu(t)), \tag{3}$$

where $\mu(t) = \mathbb{E}_t[\hat{\mathbf{x}}^{(\Omega)}(t)]$, $\mu(t_0) = \hat{\mathbf{x}}_0^{(\Omega)}$ and the *drift* is defined as $F(\mu(t)) = \sum_{k=1}^{M} \nu_k f_k(\mu(t))$. The noise term $\xi(t)$ is modeled as a zero-mean Gaussian distribution, i.e., $\xi(t) = \mathcal{N}(\mathbf{0}, C(t))$. The covariance matrix $C(t)$ evolves according to the following linear ODE:

$$\frac{dC(t)}{dt} = J(t)C(t) + C(t)J(t)^T + D(t), \tag{4}$$

where the Jacobian is defined as $J_{ij}(t) = \sum_{k=1}^{M} \nu_{ki} \partial_j f_k(\mu(t))$ and the diffusion term is defined as $D_{ij}(\mu(t)) = \sum_{k=1}^{M} \nu_{ki}\nu_{kj} f_k(\mu(t))$. These two ODEs, (3) and (4), can be numerically solved to obtain how $\mu(t)$ and $C(t)$ evolve. This results in approximating the counting process described by the CME as a Gaussian process, where, at time t, $\mathbf{x}(t)$ is a multivariate Gaussian distribution with mean $\mu(t)$ and covariance $\Omega^{-1}C(t)$, i.e. $\mathbf{x}(t) \sim \mathcal{N}(\mu(t), \Omega^{-1}C(t))$.

Representation as bipartite graph. A CRN can be graphically represented as a colored bipartite interaction graph as shown in Fig. 1 (left). Species and reactions denote two different types of nodes, represented respectively as circles and squares in Fig. 1 . The edge color denotes the stoichiometric coefficient associated with that edge: a_{ij} for edges from species to reactions and b_{ij} for edges from reactions to species. The color can be seen as a label $\{0, 1, 2, \ldots, K\}$, where $K \in \mathbb{N}$ is the largest allowed coefficient, i.e., the maximum number of individuals of a given species that can be involved in a reaction[1]. The CME cannot be solved analytically but we can sample from it using the Gillespie simulation algorithm (Gillespie, 1977) that produces stochastic trajectories that illustrate the exact time evolution of the CRN. Fig. 1 (right) shows an example of such trajectories for the CRN depicted by the bipartite graph in the middle of Fig. 1. The synthetic dataset used for training the VAE consists of a pool of trajectories sampled from the CME via Gillespie simulations, capturing the system's state at discrete equidistant time points.

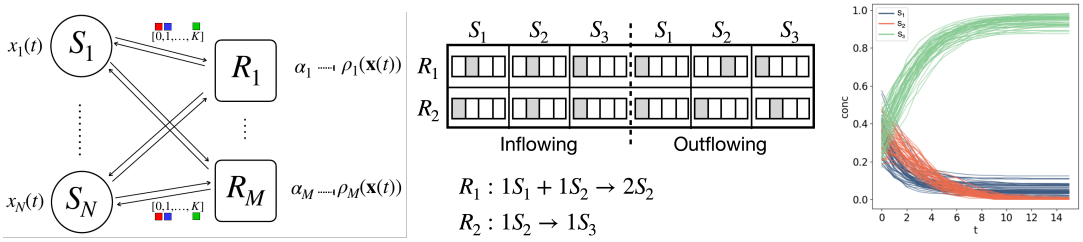

Figure 1: A CRN represented as a colored bipartite graph (left). An example of the matrix representation of a given CRN model with $N = 3$, $M = 2$ and $K = 4$ (middle). A few sampled trajectories of the CRN obtained via Gillespie simulation - our synthetic time-series data (right).

3. Methods

The goal is to infer an explicit representation of a CRN from a pool of observed realizations of an unknown MPP. The CRN structure and its dynamics are uniquely identified by the reactions $R_1, ..., R_M$, and the rate parameters $\alpha_1, ..., \alpha_M$. We leverage the colored bipartite graph formulation to encode reactions. In particular, each reaction node is univocally associated with one of the M CRN reactions. Each edge (either in- or out-flowing) is associated with a one-hot encoding of the associated stoichiometric coefficient. Therefore, the CRN structure can be identified by a $(M \times 2N \times K)$-dimensional tensor as shown in Fig. 1 (middle). Our learning framework consists of a graph-based categorical Variational Autoencoder (VAE), in which the latent space represents the CRN as bipartite interaction graph. In particular, we have a categorical distribution over these

1. The bipartite representation of CRNs resembles Petri nets, where places are species (tokens representing molecule counts), transitions are reactions, firing rates are reaction rates, and marking is a system state. CRNs are semantically equivalent to the stochastic Petri nets where transitions fire with exponentially distributed delays Heiner et al. (2008).

graphs, i.e. the one-hot encoding is replaced by a probability vector specifying the probability associated with each coefficient in each edge. Fig. 2 shows a schematic view of the VAE architecture. Let's consider a heterogeneous graph $G = (V, E)$, where $V = (V_S, V_R)$ is the set of nodes, V_S for species and V_R for reactions, and $E = (E_{S\to R}, E_{R\to S})$ is the set of edges, where $E_{S\to R}$ are edges from species to reaction nodes whereas $E_{R\to S}$ are edges from reaction to species nodes. Let $L = \{0, 1, 2, \dots, K-1\}$ be the set of possible edge types (colors), i.e., the stoichiometric coefficients of the reactions.

Dataset. The dataset consists of observed realizations of the MPP evolution at equidistant time instants $[t_0, \dots, t_T]$. Given a fixed time step Δt, $t_{j+1} = t_j + \Delta t$ for every $j \in \{0, \dots, T-1\}$. The input to the VAE is a pool of n trajectories stored in dataset $\mathbf{X} = \{\mathbf{x}^1_{0:T}, \dots, \mathbf{x}^n_{0:T}\}$, where $\mathbf{x}^i_{0:T} = (\mathbf{x}^i(t_0), \dots, \mathbf{x}^i(t_T))$, and $\mathbf{x}^i(t) = (x^i_1(t), \dots, x^i_N(t))$ denotes the state of observable species at time t. To ease the learning we work with normalized states, i.e., the concentration of individuals of a certain species over the total volume. Mathematically, the normalized data are $\hat{\mathbf{x}}^\Omega(t) := \frac{\mathbf{x}(t)}{\Omega}$.

Latent space. The latent space $Z \subseteq \{0,1\}^{|E|\times(K+1)}$ of the VAE can be visualized as space of tensors $\mathbf{z}$ of size $|E| \times K$, where $|E|$ denotes the number of edges in the bipartite graph. In its fully-connected version, $|E| = M \times 2N$. Each edge e_{ij} between nodes v_i and v_j is associated with a row in $\mathbf{z}$, representing the one-hot encoding of the associated stoichiometric coefficient (as shown in Fig. 1 (middle)). In other words, z_{ij} is a discrete categorical variable representing the edge type between v_i and v_j (one-hot representation for K edge types). For example, if the stoichiometric coefficient for edge e_{ij} is 2, then $z_{ij} = [0, 0, 1, 0, \dots]$. The VAE places a discrete distribution over this latent space, representing the likelihood of every coefficient for each edge.

Prior. The first step lies in choosing a prior distribution $p(\mathbf{z})$ over the latent space Z. For simplicity, we define it to be a factorized uniform distribution over L, i.e. over edge types, $p(\mathbf{z}) = \prod_{i=1}^{M}\prod_{i=1}^{2N} p(z_{ij})$, where $p(z_{ij}) = [\frac{1}{K}, \dots, \frac{1}{K}]$. However, one can decide that some stochastic coefficients are more likely than others. For instance, assigning a higher probability to 0 enforces sparsity. So another possible prior is $p(z_{ij}) = [2^{-1}, 2^{-2}, \dots, 2^{-K+2}, 2^{-K+1}, 2^{-K+1}]$.

Encoder. The encoder predicts the interactions given the trajectories. In practice, the encoder learning task can be framed as a (multi-label) edge classification task. It learns to classify all the $|E|$ edges simultaneously. Its architecture is a graph convolutional network (GCN) (Zhang et al., 2019) designed to output a parametric and categorical distribution $q_\phi(\mathbf{z}|\mathbf{x})$ that performs edge classification. The goal is to infer pairwise interaction types z_{ij} given observation $\mathbf{x}$. We apply the GCN on the fully connected graph to predict the latent graph structure $q_\phi(z_{ij}|\mathbf{x}) = \text{softmax}(g_{enc,\phi}(\mathbf{x})_{ij,1:T})$, where $g_{enc,\phi}(\mathbf{x})$ is a GCN passing messages over the fully connected graph (with no self-loops). Given n trajectories $\mathbf{x}^1_{0:T}, \dots, \mathbf{x}^n_{0:T}$, where $\mathbf{x}^j_{0:T} = \big(\mathbf{x}^j(t_0), \mathbf{x}^j(t_1), \dots, \mathbf{x}^j(t_T)\big)$, the encoder computes the following *message-passing* operations: for every $v \in V$ and for every $e \in E$

$$v : \mathbf{h}^1_j = g_{emb}(\mathbf{y}^j_{0:T}) \qquad e \to v : \mathbf{h}^2_j = g^1_v\Big(\sum_{i:(i,j)\in E} \mathbf{h}^1_{(i,j)}\Big)$$

$$v \to e : \mathbf{h}^1_{(i,j)} = g^1_e([\mathbf{h}^1_i, \mathbf{h}^1_j]) \qquad v \to e : \mathbf{h}^2_{(i,j)} = g^2_e([\mathbf{h}^2_i, \mathbf{h}^2_j]).$$

We can directly compute the node embedding $\mathbf{h}^1_j$ only for observable species. However, since reaction nodes are not directly observable, we embed them in learnable latent vectors sharing the same dimensionality as $\mathbf{x}^j_{0:T}$, so that the same embedding function g_{emb} can be used. In practice,

$\mathbf{y}^j_{0:T} = \mathbf{x}^j_{0:T}$ if v is a species node, and $\mathbf{y}^j_{0:T} = r_{emb}(\mathbf{x}^j_{0:T})$ if v is a reaction node. Alternatively, one can encode reactions in a different latent space and then introduce an ad-hoc node embedding function g_{remb}. Functions r_{emb} (or g_{remb}), $g_{emb}, g^1_e, g^1_v, g^2_e$ are one-dimensional convolutional neural networks (CNN) (Li et al., 2021), therefore, their inputs and outputs have fixed dimensions. The posterior distribution over edge types is computed as $q_\phi(z_{ij}|\mathbf{x}) = \text{softmax}(\mathbf{h}^2_{(i,j)})$, which is a vector of probabilities over L that sums up to one for every $i \in \{1, \dots, M\}$ and $j \in \{1, \dots 2N\}$. The parameter ϕ in the GCN $g_{enc,\phi}$ summarizes all the GCN parameters.

Remark 1 *If we do a single message-passing step in the GCN, the posterior $q_\phi(z_{ij}|\mathbf{x})$ depends only on $\mathbf{h}^1_{(i,j)}$ an in turn on $\mathbf{x}^i_{0:T}$ and $\mathbf{x}^j_{0:T}$, ignoring interactions with other nodes, while $\mathbf{h}^2_{(i,j)}$ uses information from all the graph allowing to disentangle multiple interactions.*

Sampling from the latent distribution $q_\phi(z_{ij}|\mathbf{x})$ without breaking the differentiability of the VAE computational graph requires a trick. First, we need to make a continuous approximation of the discrete latent variables so that we can use the reparameterization trick and back-propagate to obtain gradients. This trick is known as the Gumbel trick (Jang et al., 2016) and consists in sampling a vector $\mathbf{g} \in \mathbb{R}^K$ of i.i.d. samples from a $Gumbel(0, 1)$ distribution so that $z_{ij} = \text{softmax}((\mathbf{h}^2_{(i,j)} + \mathbf{g})/\tau)$, where τ is the temperature. To summarize, the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ infers the interaction graph from the trajectories in $\mathbf{X}$ assuming an initial fully-connected bipartite graph using a GCN that maps the trajectories $\mathbf{x}$ to pairwise categorical distributions z_{ij}. The resulting latent categorical distribution $q_\phi(z_{ij}|\mathbf{x}) = \text{softmax}(g_{enc,\phi}(\mathbf{x})_{ij,1:T})$ with GCN $g_{enc,\phi}(\mathbf{x})$ is an explicit representation of the distribution over possible CRN structure given the observations $\mathbf{x}$.

Remark 2 *The encoder's flexibility in learning the full graph from data, in principle, allows for some species to be unobserved. In such cases, we reconstruct missing information for the unobserved nodes using a parametric function that infers their initial state from the observed trajectories. The experiments in this paper focus only on fully observable CRNs.*

Decoder The decoder should learn the parameters governing the dynamics given the proposed interaction graph. We are considering Markovian systems so the stochastic dynamics can be written as follows:

$$p_\theta(\mathbf{x}_{0:T}|\mathbf{z}) = \prod_{t=0}^{T-1} p_\theta(\mathbf{x}(t+1)|\mathbf{x}(t), \mathbf{z}).$$

From the graph $\mathbf{z}$, we easily retrieve the stoichiometric coefficients a_{ij} and b_{ij} of each reaction and thus the related update vector ν_j. The rate functions can be expressed as polynomials of the form:

$$f_j(\mathbf{x}(t)) = \alpha_j \prod_{i=1}^{N} \binom{x_i(t)}{a_{ij}},$$

and we can easily compute the partial derivatives needed for the LNA. Thus we have an algebraic equation for the drift F, the Jacobian J and the diffusion matrix D from which we obtain $\frac{d\mu(t)}{dt}$ and $\frac{dC(t)}{dt}$. If we can solve them continuously, we have an expression for $\mu(t)$ and $C(t)$, so that

$$p_\theta(\mathbf{x}_{1:T}|\mathbf{z}, \mathbf{x}_0) = \mathcal{N}(\mu_{1:T}, \Omega^{-1} C_{1:T}).$$

Time is discretized over the time grid $[t_0, \ldots, t_T]$ (the same as our observations $\mathbf{x}$) so we can think about using the forward explicit Euler method (a.k.a. Runge-Kutta of order one) as a numerical procedure to solve ODE with a given initial value. Given $\mathbf{x}(t_0)$, we know that $\mu(t_0) = \mathbf{x}(t_0)$ and that $C(t_0) = \mathbf{0}$. In general, we could solve the ODE with a finer time discretization but this implies higher computational costs at training time. One optimal strategy would be to regulate the discretization scheme based on the kinetic rates, the higher the rate the finer the grid. The rationale is that high rates translates in fast reactions and thus in a rapidly changing dynamics. The Euler scheme produces the following iterative estimate for $k \in \{0, T-1\}$:

$$\begin{aligned} \mu(t_{k+1}) &= \mu(t_k) + \Delta t_k \cdot F(\mu(t_k)), \\ C(t_{k+1}) &= C(t_k) + \Delta t_k \cdot (J(t_k)C(t_k) + C(t_k)J(t_k)^T + D(k)). \end{aligned} \tag{5}$$

where $\Delta t_k = t_{k+1} - t_k$. Euler method is a first-order method, which means that the local error (error per step) is proportional to the square of the step size.

The decoder $p_\theta(\mathbf{z})$ is composed of a multi-layer perceptron (MLP) that maps the categorical latent distribution $q_\phi(\mathbf{z}|\mathbf{x})$ over CRN formulae into reaction rates $\alpha_1, \ldots, \alpha_M$ that fully determine the dynamics of the system.

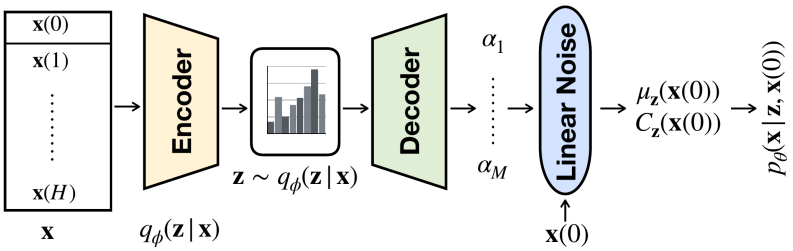

Figure 2: Diagram of the graph-VAE.

Loss. The rationale of a VAE is to minimize the Kullback-Leiber (KL) divergence (Joyce, 2011) between the true and the proposed distribution. However, as the true posterior is unknown, we derive from the KL formula a lower bound of the marginal log-likelihood of our data. This lower bound, known as the Evidence Lower Bound (ELBO) (Heard, 2021), is defined as:

$$\text{ELBO}(\theta, \phi|\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log(p_\theta(\mathbf{x}|\mathbf{z}))] - KL[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})]. \tag{6}$$

The parameters ϕ and θ that maximize the ELBO also maximize the marginal log-likelihood of data. The loss is defined as the negative ELBO, i.e., $\mathcal{L}(\theta, \phi|\mathbf{x}) = -\text{ELBO}(\theta, \phi|\mathbf{x})$. The first term, known as *reconstruction error*, can be estimated as the negative log-likelihood of our data w.r.t. the multivariate Gaussian resulting from the LNA:

$$\sum_{t=1}^{T} \frac{(\mathbf{x}(t) - \mu(t))^T C(t)^{-1} (\mathbf{x}((t) - \mu(t))}{2} + \frac{1}{2} \det\big(C(t)\big).$$

The second term, known as *regularization term*, is added to the first and computes the KL divergence between the proposed posterior, $q_\phi(z_{ij}|\mathbf{x})$, and the prior distribution, $p_\theta(\mathbf{z})$, over the discrete latent space Z. The KL term for a uniform prior and a softmax distribution is the sum of entropies:

$$\sum_{i \neq j} H(q_\phi(z_{ij}|\mathbf{x})) = \sum_{i \neq j} \sum_{k=1}^{K} q_\phi(z_{ij}|\mathbf{x})(k) \cdot [\log\left(q_\phi(z_{ij}|\mathbf{x})(k)\right)) - \log\left(p(z_{ij}(k))\right)].$$

We can leverage automatic differentiation tools to compute gradients of the loss via backpropagation and perform gradient-based optimization.

Training and Evaluation. For each trajectory, we sample a CRN model from the latent distribution and compute the LNA of the dynamics. Training is performed by optimizing the weights of the VAE through backpropagation. At test time, the *most likely CRN graph* is used together with the resulting reaction rates to make predictions.

Identifiability. In general, different CRN graphs may lead to similar dynamics (Craciun and Pantea, 2008). To tackle this, we introduce the measure of likelihood of data w.r.t. the LNA multivariate Gaussian approximation of the inferred networks. In practice, this measure can serve as a tool for model selection. Given that we use the LNA approximation for the models, our methodology will not differentiate between models that are equally distant from the true distribution. A detailed study of the identifiability of CRNs from time-series data is subject to future work. For now, we introduce *CRN divergence*, a measure that quantifies how much the inferred model differs from the true model when the latter is known. We define it as the KL divergence between LNA Gaussian distributions of the inferred and the true model.

4. Experiments

To validate the proposed method, we conduct experiments on synthetic model-based trajectories. We generate multiple initial states and trajectories by simulating from a known ground-truth model using the aforementioned Gillespie algorithm. In this implementation, we assume prior knowledge of the number of species and reactions in the system. However, in a more general setting, one could allow for additional species and reactions, enabling the algorithm to infer the most likely system that explains the given data. To enhance training stability, we implement an alternating training scheme in which one network is frozen while updating only the other. Specifically, we train the encoder alone for 10 epochs, followed by training the decoder alone for 5 epochs, alternating throughout. Additionally, to prevent the network from rapidly converging to suboptimal solutions we scale the activation functions of the final layer of the encoder and decoder (sigmoid or softmax) to obtain a slower saturation.

In the following, we demonstrate the applicability of our framework to three case studies of increasing complexity. For each experiment, we first show the true model, then outline the training and testing parameters, and finally present the inferred CRN. Table 1 summarizes key experimental results of each experiment. To illustrate the quality of inference, Fig. 3 compares trajectories generated by the ground-truth model (solid lines) with the dynamics of our inferred model, represented as the LNA mean (dashed lines).

One-reaction network. The first case study considers a simple network consisting of a single reaction, where species A transforms into species B at rate $\alpha_1 = 0.2$. The true CRN is given by:

$$R_1 : 1A \xrightarrow{0.2} 1B.$$

We simulated this system with a total volume of $\Omega = 200$, generating training data consisting of 1500 trajectories. These were obtained from 150 randomly sampled initial states, with 10 trajectories per initial state. Time-series measurements were collected over a time horizon of $T = 11$ time units, with data samples taken at intervals of $\Delta t = 0.5$ time units. The graph-based VAE was

trained as described in Section 3 and validated on an independent dataset of 1000 trajectories from 100 randomly sampled initial states. After training, the VAE converged to the correct network with rate 0.1904. This inferred model retains the same reaction structure as the true model but exhibits a slightly slower reaction rate. Fig. 3 (left) visualizes sample trajectories from the ground-truth model, where species A is shown in blue and species B in red. The close alignment of the inferred trajectories indicates that the model effectively captures the system's dynamics. Table 1 reports a KL divergence of 0.0718, quantifying how far the inferred model is from the true model. The low KL divergence demonstrates the accuracy of the inference result, indicating that the inferred CRN is satisfactory with respect to the data.

Birth-death. Next, we analyze a simple birth-death model, where species A converts to species B at rate α_1, while species B transforms back into species A at a higher rate α_2. The true CRN is given by:

$$R_1 : 1A \xrightarrow{0.25} 1B,\ 1B \xrightarrow{0.75} 1A.$$

As the complexity of the system increases, we expand the training dataset to 5000 simulated trajectories from 100 randomly sampled initial states.The parameters for time steps, data points, and total volume remain unchanged: $\Delta t = 0.5$, $t = 21$, and $\Omega = 200$. The validation set consists of 1000 trajectories generated from 10 different initial states. The VAE converged to the correct network with rate 0.1968 for reaction R_1 and 0.5855 for reaction R_2. Although the inferred reaction rates are slightly lower than those in the true CRN, the overall system dynamics are captured well. Fig. 3 (middle) illustrates sample trajectories, highlighting the strong alignment between the inferred and true dynamics. Furthermore, the KL divergence of 0.1394 indicates a reasonable approximation of the true system, further demonstrating the effectiveness of the proposed inference method.

Cascade. In the final experiment, we infer a cascade model, where species A transforms into species B, and B further transforms into species C. The true CRN is defined as:

$$R_1 : 1A \xrightarrow{0.7} 1B,\ 1B \xrightarrow{1.3} 1C.$$

For this experiment, we generated a training dataset consisting of 7500 simulated trajectories with 150 different initial states, and a time horizon of $T = 8$ time units, while keeping the other parameters unchanged. The validation set includes 5000 trajectories with 100 different initial states. After 50 epochs of training, the VAE converged to the following inferred network

$$R_1 : 1B \xrightarrow{0.6664} 1C,\ R_2 : 1A \xrightarrow{0.7854} 1C.$$

Here, the inferred network structure differs from the true CRN, as the model bypasses the intermediate step of transforming A to B, instead directly converting A into C. Despite this structural difference, the inferred dynamics closely match those of the true model, as illustrated in Fig. 3 (right). This case study highlights a fundamental challenge in causal inference: distinguishing between direct causation and an indirect effect caused by a common intermediary. This issue frequently arises in biological systems, where different mechanistic explanations can produce nearly identical observed dynamics. Our result suggests that the VAE struggles with identifiability in this case - when multiple CRNs generate similar behaviours, the method may not infer the correct reaction graph.

This result underscores the challenge of identifiability: when different CRNs yield similar behaviours, the VAE may not infer the true model but can still infer a valid approximation that explains the data very well.

Experiment	Training Trajectories	Epochs	Training Time	CRN Divergence
One-reaction	1500	50	1:13h	0.0718
Birth-death	5000	50	5:19h	0.1394
Cascade	7500	50	25h	15.8102

Table 1: Training results for all three experiments.

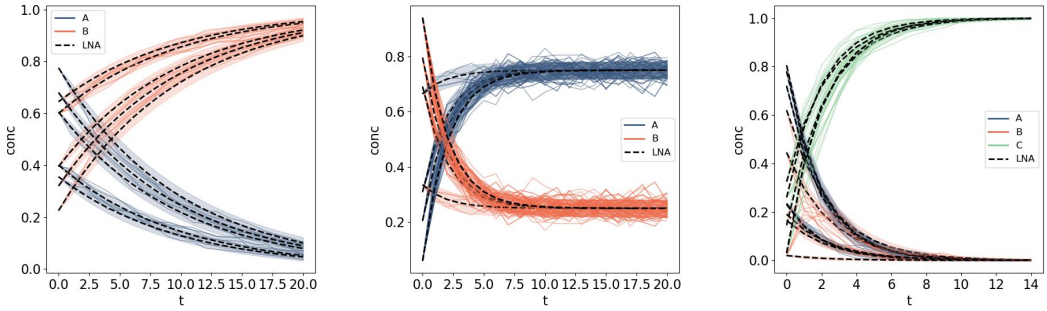

Figure 3: Simulated trajectories (colored) compared to inferred dynamics, represented by the LNA mean (black dashed lines), for randomly sampled initial states. The shaded area represents the LNA variance around the mean. Experiments: one-reaction network (left), birth-death (middle), cascade model (right).

5. Conclusion

We proposed how to automatically learn the explanatory models written in form of Chemical Reaction Networks (CRNs), from time-series measurements of a Markov population chain. We used a symbolic representation of a CRN as a bipartite graph, and we employed a graph-based Variational Autoencoder (VAE) to jointly infer the interactions and the accompanying kinetic parameters. Our contribution represents a proof-of-concept that the microscopic interactions between agents can be inferred from macroscopic, population-level data in a fully automated and data-driven fashion. Our framework is applicable both in scenarios where data is available via experiments, or when it is generated by a black-box emulator. As such, it is relevant for applications in fields ranging from epidemiology (Crepey et al., 2022), swarm robotics (Hamann et al., 2016), all the way to emerging areas such as explainable AI (Rudin, 2019) and precision medicine (Martinelli et al., 2021).

We aim at extending the framework towards automatically inferring more complex CRNs. We are actively investigating the scalability and identifiability bottlenecks of the proposed method. Increasing the number of species, reactions, or possible stoichiometric coefficients significantly expands the space of possible graphs and the dimensionality of the latent space, making the training procedure more challenging. Nonetheless, the flexibility of our framework in terms of graph representation allows us to incorporate prior knowledge by fixing parts of the graph before learning, which reduces the number of dimensions for more complex CRNs. If either the structure (mechanisms) or the rates are known, one part of the VAE can be fixed (Encoder or Decoder), allowing the remaining network to infer the missing information. To improve performance, we intend to incorporate attention mechanisms to focus on specific parts of the input. To better preserve the inherent ordering information of coefficients, we are investigating the possibility of modelling the latent space using an ordinal representation rather than a categorical one. In the future, we want to demonstrate the power and effectiveness of our framework on case studies typical for modeling in epidemiology and systems biology.

Acknowledgments

FC was partially supported by the PNRR project iNEST (Interconnected North-Est Innovation Ecosystem) funded by the European Union Next-GenerationEU (Piano Nazionale di Ripresa e Resilienza (PNRR) – Missione 4 Componente 2, Investimento 1.5 – D.D. 1058 23/06/2022, ECS_00000043). JK and TP were partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2117 – 422037984.

References

Saima Absar and Lu Zhang. Discovering time-invariant causal structure from temporal data. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 2807–2811, 2021.

Aubrey Barnard and David Page. Causal structure learning via temporal markov networks. In *International Conference on Probabilistic Graphical Models*, pages 13–24. PMLR, 2018.

Luca Bortolussi and Francesca Cairoli. Bayesian abstraction of markov population models. In *Quantitative Evaluation of Systems: 16th International Conference, QEST 2019, Glasgow, UK, September 10–12, 2019, Proceedings 16*, pages 259–276. Springer, 2019.

Luca Bortolussi and Luca Palmieri. Deep Abstractions of Chemical Reaction Networks. In *International Conference on Computational Methods in Systems Biology*, volume 11095, pages 21–38, Cham, 2018. Springer. ISBN 978-3-319-99428-4 978-3-319-99429-1. URL http://link.springer.com/10.1007/978-3-319-99429-1_2.

Neil R Bramley, Tobias Gerstenberg, Ralf Mayrhofer, and David A Lagnado. Time in causal structure learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 44(12): 1880, 2018.

Francesca Cairoli, Fabio Anselmi, Alberto d'Onofrio, and Luca Bortolussi. Generative abstraction of markov population processes. *Theoretical Computer Science*, 977:114169, 2023.

Zhixing Cao, Rui Chen, Libin Xu, Xinyi Zhou, Xiaoming Fu, Weimin Zhong, and Ramon Grima. Efficient and scalable prediction of stochastic reaction–diffusion processes using graph neural networks. *Mathematical biosciences*, 375:109248, 2024.

Luca Cardelli, Milan Češka, Martin Fränzle, Marta Kwiatkowska, Luca Laurenti, Nicola Paoletti, and Max Whitby. Syntax-guided optimal synthesis for chemical reaction networks. In *Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part II 30*, pages 375–395. Springer, 2017.

Siya Chen, HaoTian Wu, and Guang Jin. Causal structure learning for high-dimensional non-stationary time series. *Knowledge-Based Systems*, 295:111868, 2024.

Matthew Cook, David Soloveichik, Erik Winfree, and Jehoshua Bruck. Programmability of chemical reaction networks. In *Algorithmic bioprocesses*, pages 543–584. Springer, 2009.

Gheorghe Craciun and Casian Pantea. Identifiability of chemical reaction networks. *Journal of Mathematical Chemistry*, 44(1):244–259, 2008.

Pascal Crepey, Harold Noël, and Samuel Alizon. Challenges for mathematical epidemiological modelling. *Anaesthesia, Critical Care & Pain Medicine*, 41(2):101053, 2022.

Daniel T Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25):2340–2361, 1977.

Ankit Gupta, Christoph Schwab, and Mustafa Khammash. DeepCME: A deep learning framework for computing solution statistics of the chemical master equation. *PLOS Computational Biology*, 17(12):e1009623, December 2021. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1009623. URL https://dx.plos.org/10.1371/journal.pcbi.1009623.

Heiko Hamann, Yara Khaluf, Jean Botev, Mohammad Divband Soorati, Eliseo Ferrante, Oliver Kosak, Jean-Marc Montanier, Sanaz Mostaghim, Richard Redpath, Jon Timmis, et al. Hybrid societies: challenges and perspectives in the design of collective behavior in self-organizing systems. *Frontiers in Robotics and AI*, 3:14, 2016.

Nick Heard. *An introduction to Bayesian inference, methods and computation*. Springer, Cham, 2021.

Monika Heiner, David Gilbert, and Robin Donaldson. Petri nets for systems and synthetic biology. In *International school on formal methods for the design of computer, communication and software systems*, pages 215–264. Springer, 2008.

Christina Heinze-Deml, Marloes H Maathuis, and Nicolai Meinshausen. Causal structure learning. *Annual Review of Statistics and Its Application*, 5(1):371–391, 2018.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

James M. Joyce. *Kullback-Leibler Divergence*, pages 720–722. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-04898-2. doi: 10.1007/978-3-642-04898-2_327. URL https://doi.org/10.1007/978-3-642-04898-2_327.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural Relational Inference for Interacting Systems. 2018.

Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12):6999–7019, 2021.

Zijian Li, Yifan Shen, Kaitao Zheng, Ruichu Cai, Xiangchen Song, Mingming Gong, Zhengmao Zhu, Guangyi Chen, and Kun Zhang. On the identification of temporally causal representation with instantaneous dependence. *arXiv preprint arXiv:2405.15325*, 2024.

Yan Liu, Alexandru Niculescu-Mizil, Aurelie C Lozano, and Yong Lu. Learning temporal causal graphs for relational time-series analysis. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 687–694, 2010.

Julien Martinelli, Sandrine Dulong, Xiao-Mei Li, Michèle Teboul, Sylvain Soliman, Francis Lévi, François Fages, and Annabelle Ballesta. Model learning to identify systemic regulators of the peripheral circadian clock. *Bioinformatics*, 37(Supplement_1):i401–i409, 2021.

Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*, 2019.

Kaan Öcal, Ramon Grima, and Guido Sanguinetti. Parameter estimation for biochemical reaction networks using wasserstein distances. *Journal of Physics A: Mathematical and Theoretical*, 53 (3):034002, 2019.

Abdellah Rahmani and Pascal Frossard. Causal temporal regime structure learning. *arXiv preprint arXiv:2311.01412*, 2023.

Denis Repin and Tatjana Petrov. Automated deep abstractions for stochastic chemical reaction networks. *Information and Computation*, 281:104788, 2021.

Benjamin M Rottman and Frank C Keil. Causal structure learning over time: Observations and interventions. *Cognitive psychology*, 64(1-2):93–125, 2012.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

F. Scarselli, M. Gori, Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini. The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1):61–80, January 2009. ISSN 1045-9227, 1941-0093. doi: 10.1109/TNN.2008.2005605. URL http://ieeexplore.ieee.org/document/4700287/.

Abhyudai Singh and Ramon Grima. The Linear-Noise Approximation and moment-closure approximations for stochastic chemical kinetics. 2017. doi: 10.48550/ARXIV.1711.07383. URL https://arxiv.org/abs/1711.07383.

Simon Valentin, Neil R Bramley, and Christopher G Lucas. Learning hidden causal structure from temporal data. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 42, 2020.

Mingjian Wen, Evan Walter Clark Spotte-Smith, Samuel M Blau, Matthew J McDermott, Aditi S Krishnapriyan, and Kristin A Persson. Chemical reaction networks and opportunities for machine learning. *Nature Computational Science*, 3(1):12–24, 2023.

Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 6(1):1–23, 2019.