Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Computational and Systems Biology

Model inference from protein time-course in Hematopoietic Stem Cells (HSC)

Pandu Raharja [1,2,3,*], Rene Schoeffel [1,2,3], Michael Strasser [3] and Carsten Marr [3,*]

[1] Technische Universität München, Fakultät für Informatik, Boltzmannstr. 3, 85748 Garching bei München, Germany
[2] Ludwig-Maximilians-Universität München, Professor-Huber-Platz 2, 80539 München, Germany
[3] Helmholtz Zentrum München, Institute of Computational Biology, Ingolstädter Landstr. 1 85764 Neuherberg, Germany.

* To whom correspondence should be addressed.

Associate Editor: Jan Quell

Received on 23/09/2016; revised on 14/10/2016; accepted on 21/10/2016

Abstract

Motivation: Stochasticity of gene expression becomes apparent when studying the dynamics of single cells rather tan a population of cells. These fluctuations in gene expression in fact reveal more information about the underlying mechanisms of transcription and translation than one could obtain from population averages.
Results: In this paper we developed a particle filtering algorithm to infer the parameters of stochastic models from single cell time series data. In particular, we apply this method to time-lapse microscopy data of two transcription factors (`Pu.1` and `Gata.1`) in murine blood stem cells. These transcription are thought to play a major role in stem cell differentiation. Our results provide several insights into the dynamics of blood stem cells maturation and specifically in the single cell environment, we managed to gain several valuable insights on the interaction dynamics between two transcription factors and the outcome of cell maturation.
Availability: The data are available upon request. A general framework for determining the parameters that best explain the data were published under open source license. The complete source code of the program is publicly accessible at https://github.com/raharjaliu/PFInfer.
Contact: pandu.raharja@tum.de
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The advances in gene expression measurement techniques, coupled with equally rapid advances in single cell analytics, have allowed the more recent single cell analysis of cell dynamics. This has enabled us to look deeper into the dynamics of stem cell maturation. In recent years, we have discovered several genes that potentially are involved in the decision making mechanism of cell maturation going from Inner Mass Cell (IMC) all the way to somatic cells (Graf & Enver, 2009).

In this paper we focus on the maturation lines of hematopoietic stem cells (Orkin *et al*, 2008). The lines start with Long Term Hematopoietic Stem Cells (LT-HSC), which is progenitor of all blood cells. During its life, LT-HSC could either divide into two LT-HSC or turn into Short Term Hematopoietic Stem Cell (ST-HSC). SH-HSC will then undergo a transition process into either Common Myeloid Progenitor (CMP) or Common Lymphoid Progenitor (CLP). As the names suggest, both cells are the ancestor of all myeloid cells and lymphocites, respectively. Each would then further maturate into their respective intermediate cells and eventually the mature somatic cells (Red Blood Cells, Megakaryocytes, Mast Cells, Eosinophils, Neutrophils and Macrophages from CMP and B and T Lymphocytes from CLP; see Figure 1).

The decision process of CMP transitioning into either MEP and GMP stands in focus of our project. Two transcription factors, `Pu.1`and `Gata.1` are thought to influence the decision process (Graf & Enver, 2009). Moreover, both transcription factors are known to inhibit each other. It is thought that the whole cross inhibition dynamics between the two has an impact on the fate of Common Myeloid Progentir. We assume this interaction to have "tipping a balance impact" on the fate of the cells, in

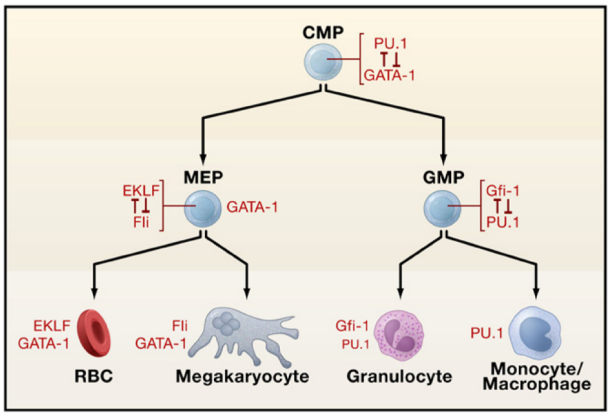


Fig. 1. Maturation cascade of intermediate Common Myeloid Progenitor (CMP) to Megakaryocyte-Eerythroid Progenitor (MEP) and Granulocyte/Macrophage Progenitor (GMP). MEP would then maturate into somatic Red Blood Cells and Megakaryocites while GMP would undergo maturation into Granulocytes and Monocyptes. For each process there are known factors which are known to interact antagonistically on maturation decision. Our CMP to MEP and GMP decision stands on the top of the figure with two transcription factors of interest, `Pu.1` and `Gata.1` interact atagonistically.

which a dominating effect of one transcription factor would have a decisive role on the transition.

For this project we use particle filtering to learn more about this dynamics. We apply this technique on our time-dependent expression data of single cells containing concentration of both transcription factors. The approach of the experiment would be further described in the next section followed by the theoretical aspects of the particle filtering and other methods used in this project.

2 Approach

For the single cell dynamic analysis to be possible, we were interested in gaining insight in single cell transcription data of both `Pu.1` and `Gata.1`. We combine two techniques for this to be possible: single cell tracking and protein tagging.

2.1 Experimental Setting

The stem cells were put in a culture for one week. Two videos of the cells, taken under bright-field and ultraviolet microscopy, would then be taken for the whole duration of experiment. An automatic detection program tracks down each cell uniquely within the video frame. The program also tracks down cell division would track the newly created cells likewise. It also detects cell death.

Once every 20-30 minutes a protein expression levels of each cell would then be measured. The expressed protein of interests (in our case `Pu.1` and `Gata.1`) were tagged with distinct fluorescence proteins, which would allow for the expression level measurement at each time point.

3 Models

We model our

3.1 Particle Filtering

3.1.1 Particle

We use particle filtering in our simulation. Particle filtering is a family of methods that use utilize the concept of *particle*. A particle $\mathcal{K}$ is defined as a triple of trajectory X, parameter set θ and assumed model $\mathcal{M}$,

$$\mathcal{K} := (X, \theta, \mathcal{M}) \tag{1}$$

Specifically in our case, trajectory refers to the concentration of measured proteins `Pu.1` and `GATA1` while parameter set encompasses all variables that were involved in the reaction such as propensity and its corresponding auxiliary variables (see subsection **Reaction Models**). Note that a trajectory is different from the experimental data $\mathcal{D}$. A trajectory is the result of the simulation done by applying the parameters onto our model. In functional notation we would assume the i-th value of the trajectory X_i to be a function of the i-th value of the trajectory, the model and the parameters,

$$X_i = f(X_{i-1}, \mathcal{M}, \theta_{i-1}) \tag{2}$$

3.1.2 Posterior

Our posterior describes the probability of having the trajectory X and parameter θ given the observation $\mathcal{D}$,

$$P(X, \theta|\mathcal{D}) \tag{3}$$

We could understand this as the probability of having our simulation return a given set of values *and* having the parameter set θ given that we previously observed the experimental data $\mathcal{D}$. Using Bayes' Theorem we could further expand our posterior into an update rule,

$$P(X, \theta|\mathcal{D}) = \frac{P(\mathcal{D}|X, \theta)P(X, \theta)}{P(\mathcal{D})} \tag{4}$$

In our simulation we know that, to compute prior $P(\mathcal{D}|X, \theta)$, only knowledge about the trajectory of the simulation is needed. Hence, we could simplify our prior,

$$P(\mathcal{D}|X, \theta) = P(\mathcal{D}|X) \tag{5}$$

Not that the above equation inherently assumes that $\mathcal{D}$ is only directly dependent on X and X is in turn only directly dependent on θ. Incorporating this onto our update rule, and expanding the definition of $P(X, \theta)$ using chain rule, we get

$$P(X, \theta|\mathcal{D}) = \frac{P(\mathcal{D}|X)P(X|\theta)\pi(\theta)}{P(\mathcal{D})} \tag{6}$$

One of the interesting aspects of our is the fact that the i-th simulation result is only dependent on previous simulation, a property known as *Markov property*. We could thus rewrite the update rule as follow,

$$P(X, \theta|\mathcal{D}) = \frac{\prod_{i=0}^{N} P(\mathcal{D}_i|X_i)P(X_0)\prod_{l=1}^{N} P(X_{[t-1,ti]}|X_i, \theta)\pi(\theta)}{P(\mathcal{D})} \tag{7}$$

3.1.3 Parameters

During the simulation we assumes certain parameters that would influence the trajectory of the simulation. The parameter set θ could then mathematically be understood as P-tuple containing all the parameters that are assumed in the simulation,

$$P := (P_0, P_1, \ldots, P_P) \tag{8}$$

3.1.4 Prior

Our prior $\pi(\theta)$ could be expanded by assuming the independence of each parameter θ_i within the parameter set θ,

$$\pi(\theta) = \prod_{i=1}^{P} \pi(\theta_i) \tag{9}$$

Specifically for this simulation, we assume our prior to be Gamma distributed with the parameters α and β,

$$\pi(\theta) = \prod_{i=1}^{P} Ga(\theta_i, \alpha_i, \beta_i) \tag{10}$$

We used Gamma distribution as our prior due to the fact that a conjugate prior of a Gamma distributed random variable is also Gamma distributed.

3.2 Simulation Process

The simulation is run in roughly following high level steps:

1. Initialization of parameters θ.
2. Input of data $\mathcal{D}$.
3. Particle filtering routine:

   a. Generation of initial particles for step i

   $$Ki := (K_{i1}, K_{i2}, \ldots, K_{im}) \tag{11}$$

   b. Simulation run of each particle K_{ij}
   c. Weighting of each particle. The weight is a function of probability of observing the data given the simulation result.

   $$w_i^k = P(D_i|X_i^k) = \mathcal{N}(D_i|X_i^k) \tag{12}$$

   d. Parameter update for every K,

   $$\theta^k \propto P(\theta|X^k_{[to,ti]}) \tag{13}$$

4. Model comparison. This could be done by measure such as AIC, BIC, Bayes Factor etc.

3.2.1 Particle Generation

As mentioned before, A particle $\mathcal{K}$ is defined as a 3-tuple of trajectory X, parameter set θ and assumed model $\mathcal{M}$. For a specific model, j-th particle at i-th time particle is then just a tuple of trajectory and parameter set $k_{ij} = (x_i, \theta_{ij})$.

For initial time, a particle could be constructed by combining initial parameters, which were arbitrarily defined by a pre-defined prior, and initial trajectory value X_0. There are three ways to define the initial trajectory value. First is to take 0 as initial value. Second is to take the first measurement data $\mathcal{D}_0$. Third is to model initial data as normal distributed instances around $\mathcal{D}_0$, i.e. $X_0 \propto \mathcal{N}(\mathcal{D}_0)$. It is worth noting that the first strategy may not always be applicable.

M particles would then be created and for each iteration of the simulation with each particle having weight which calculated from previous iteration w_{i-1}^k. In the next iteration, M particles are to be chosen. Note that this implies that a particle from previous iteration could be chosen more than once. Corollarly, a particle from previous iteration may also not be chosen at all. The draw is done by first drawing a random $z \propto \mathcal{U}(0,1)$. A particle k_{ij} is chosen with j denoting the smallest index so that the sum of all weight of particles with index smaller than j is larger than z, i.e.

$$\sum_{l=1}^{j} w_{i-1}^l \geq z \tag{14}$$

3.2.2 Simulation Run

Simulation is done using Gillespie's SSA algorithm. There are possible improvements that we thought might be useful in our project such as τ-leaping and Bayesian Approximation Method. Such improvements are generally essential since it would reduce computational cost of the simulation and in turn enable the simulation to scale further to accommodate more leveraged simulation – in our case we could simulate more cells at the same time in possible more fine-grained time-resolved manner. In our case especially, some kind of improvement would be needed to accommodate the explosion of the number of cells within our culture. Without some kind of approximative method there would be a bottleneck in simulation caused by the exponential increase of the number of living cells that have to be simulated.

Tau Leaping

Generally, tau-leaping works by performing all reactions happening for an interval of length tau before updating the parameters and propensity function. By allowing this kind of approximation we potentially allow more efficient simulation and thus increase the capability of the simulation to cope with larger systems. In our case, we use following tau,

$$\tau = \frac{1}{a_0} ln\Gamma \tag{15}$$

with a_0 referring to total sum of reaction propensities and Γ being uniformly distributed between 0 and 1.

3.2.3 Particle Weighting

Upon the completion of all simulations within one iteration, the quality of each simulation (and in turn, the particle) will be quantified. This quantification is implied in the weighting of the particle at the next iteration w_i^k. We assume the Gaussian distribution of particle around the measurement time at time $t = i$. Using this assumption, we can quantify our weight in following manner,

$$w_i^k = P(\mathcal{D}_i|X_i^k) = \mathcal{N}(\mathcal{D}_i|X_i^k) \tag{16}$$

3.2.4 Parameters Update and Gamma Distribution

After each run, the parameters would then be updated using Gamma Distribution. For k-th particle, the updated parameters at time i are dependent on previous parameters given the trajectory of data in $[t_0, t_i]$,

$$\Theta^k \propto P(\Theta|X_{[t_0,t_i]}) \tag{17}$$

Applying Bayes' Rule on the probability and independent assumption, we get following equation,

$$P(\Theta|X) = \frac{P(X|\Theta)}{P(X)} \prod_i P(\Theta_i) \tag{18}$$

Gamma Distribution is particularly favored for the update since a conjugate prior of Gamma Distribution is in turn Gamma distributed,

$$P(\Theta|X) = \frac{P(X|\Theta)}{P(X)} \prod_{d=1}^{P} Ga(\Theta_d, \alpha_d, \beta_d) \tag{19}$$

With α and β referring to both the shape and rate of the distribution respectively. Owing to the fact that conjugate prior of Gamma Distribution is also Gamma distributed, this could be then further simplified as an update rule of the Gamma distribution,

$$P(\Theta|X) = \frac{P(X|\Theta)}{P(X)} \prod_{d=1}^{P} Ga(\Theta_d, \alpha_d + r_d, \beta_d + Gd) \tag{20}$$

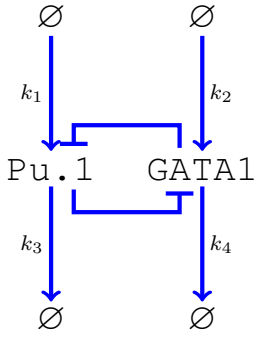
r_d here refers to the number of times the d-th reaction was fired during the run and G_d is defined as follows,

$$G_d := \frac{1}{k_d} \sum_{l=0}^{i} a_l(X(S)) \tag{21}$$

Here, k_d and $a_l(X(S))$ refer to reaction rate of the d-th reaction and trajectory-dependent propensity function, respectively.

3.3 Reaction Models

We assume following putative cross inhibition between genes `Pu.1` and `GATA1` playing roles in cell differentiation dynamics:



There are four possible reactions happening within our model: k_1, k_2, k_3 and k_4. In our simulation, each reaction was assigned propensity value which roughly is proportional to the likelihood of the reaction happening in a given time:

$$P(R_i = k_l) \propto a_l \tag{22}$$

The propensity a of decay reactions k_3 and k_4 roughly follows the mass action law of chemical reaction:

$$a_l = k_l \cdot A; l \in [3, 4] \tag{23}$$

For production reactions k_1 and k_2 we assume Michaelis-Menten inhibition kinetics that influences the production of the species. In this assumption, the inhibiting characteristics of a species on another species negatively influences the creation rate of the species it inhibits. The propensity of the reactions are thus,

$$a_1 = k_1 \cdot \frac{\texttt{GATA1}^{-n}}{\texttt{GATA1}^{-n} + K_{P1}^{-n}} \tag{24}$$

$$a_2 = k_2 \cdot \frac{\texttt{Pu.1}^{-n}}{\texttt{Pu.1}^{-n} + K_{P2}^{-n}} \tag{25}$$

3.4 Models Comparison

Our framework enables us to test not only standard hematopoietic stem cell maturation as described above, it also allows us to compare it with other models. One method that could be used to compare models is Bayes Factors. It is done by performing ratio of the posterior probabilities of two models M_1 and M_2,

$$B_{M1,M2} = \frac{P(M_1|\mathcal{D})}{P(M_2|\mathcal{D})} \tag{26}$$

Using Bayes' Theorem, we can formulate the marginal probability as,

$$P(M|\mathcal{D}) = \frac{P(\mathcal{D}|M)P(M)}{P(\mathcal{D})} \tag{27}$$

The marginal likelihood of model $P(\mathcal{D}|M)$ could be approximated using the particles at each iteration i (Wilkinson, 2011). Since the observation of particles only depends on the previous particle, We could therefore rewrite the term as,

$$P(\mathcal{D}|M) = P(\mathcal{D}_\prime) \prod_{l=1}^{N} P(\mathcal{D}_{i+1}|\mathcal{D}_{0:i}, M) \tag{28}$$

Assuming a priori equally likely models, the factor of $P(M)$ in (26) cancels between the two models and the Bayes Factors now becomes the ratio of two marginal likelihood,

$$B_{M1,M2} = \frac{P(\mathcal{D}|M_1)}{P(\mathcal{D}|M_2)} \tag{29}$$

4 Discussion

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam quam sapien, gravida quis blandit eget, euismod sit amet ante. Ut sapien sem, tincidunt ut sollicitudin ut, fermentum eu sapien. Quisque interdum aliquet ante non luctus. Donec ex purus, vehicula et pellentesque eu, tempus in tellus. Nullam euismod nisl vitae enim dignissim finibus. Maecenas laoreet risus in placerat auctor. Phasellus id magna vitae metus placerat porttitor at a nibh. Aliquam sit amet tortor ut lorem feugiat gravida. Interdum et malesuada fames ac ante ipsum primis in faucibus. Nulla venenatis erat id dolor pretium, ac lacinia odio elementum. Nunc non imperdiet tortor.

5 Conclusion

Equation (1) Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text. Figure 2 shows that the above method Text Text Text Text Text Text Text Text Text Text Text Text. **?** might want to know about text text text text

Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text. Figure 2 shows that the above method Text Text Text Text

Acknowledgements

Text Text Text Text Text Text Text Text. **?** might want to know about text text text text

Funding

This work has been supported by the... Text Text Text Text.

References

Feigelman, J. (2011). "Stochastic and deterministic methods for the analysis of Nanog dynamics in mouse embryonic stem cells." PhD Thesis, Technische Universität München, Munich, Germany.

Graf, T., & Enver, T. (2009). Forcing cells to change lineages. *Nature*, **462(7273)**, 587-594.

Filipczyk, A., Marr, C., Hastreiter, S., Feigelman, J., Schwarzfischer, M., Hoppe, P. S., ... & Hilsenbeck, O. (2015). Network plasticity of pluripotency transcription factors in embryonic stem cells. *Nature cell biology.*

Orkin, S. H., & Zon, L. I. (2008). Hematopoiesis: an evolving paradigm for stem cell biology. *Cell*, **132(4)**, 631-644.

Gillespie, D. T. (2007). Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, **58**, 35-55.

Fahrmeir, L., Künstler, R., Pigeot, I., & Tutz, G. (2007). *Statistik: Der Weg zur Datenanalyse*. Springer-Verlag.

Wilkinson, D. J. (2011). Stochastic modelling for systems biology. CRC press.

Zechner, C., Pelet, S., Peter, M., & Koeppl, H. (2011, December). Recursive Bayesian estimation of stochastic rate constants from heterogeneous cell populations. In *2011 50th IEEE Conference on Decision and Control and European Control Conference* (pp. 5837-5843). IEEE.

Haseltine, E. L., & Rawlings, J. B. (2005). On the origins of approximations for stochastic chemical kinetics. *The Journal of chemical physics*, **123(16)**, 164115.

Sherlock, C., Golightly, A., & Gillespie, C. S. (2014). Bayesian inference for hybrid discrete-continuous stochastic kinetic models. *Inverse Problems*, **30(11)**, 114005.