Causal DAG for genes obtained via Mappa Mundi algorithm

Robert R. Tucci
tucci@ar-tiste.com

March 15, 2025

Abstract

1 Introduction

This paper can be viewed as an application and further refinement of the Mappa Mundi (MM) algorithm. In this case, we apply it to finding what are called Gene Regulatory Networks (GRN), autoregulon (AR) nets and Network Motifs in the Genomics and Systems Biology literature (Ref.[1]).

The MM algorithm was first proposed in Ref.[4] for DAG Extraction From Text (DEFT). In Ref.[4], it was used to compare 3 P.G. Wodehouse short stories and the scripts of 3 PiXar movies and extract from those causal DAGs. In general, the MM algorithm can extract causal DAGs from 2 or more text files, as long as each of those text files recounts actions in chronological order. So it will work with time stamped lab notebooks or medical records, but it won't work with textbooks or fiction with time travel or flashback shenanigans.

After Ref.[4], the MM algorithm was later applied in Ref.[3] to extracting causal DAGs from FitBit times series data.

In this paper, we use the MM algorithm to extract DAGs from time series data for concentrations of gene expressions and transcription factors. The DAGs we obtain are called GRN in the Genomics literature. GRN are a special case of AR networks.

AR nets are discussed in the chapter entitled "Autoregulon Networks (Network Motifs)" of my book Ref.[2].

AR nets are a special case of Dynamical systems (DS). DS are discussed in the chapter entitled "Dynamical Systems" of my book Ref.[2].

2 Comparing 2 TS Records

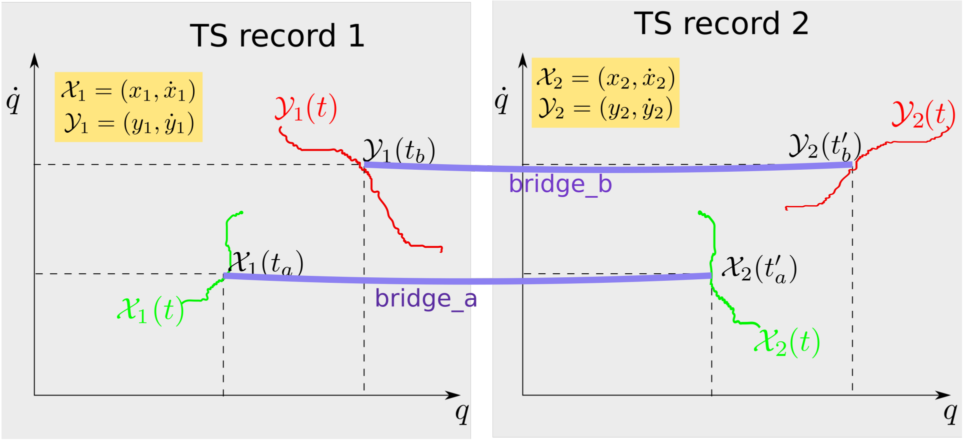


Figure 1: Causal bridges a and b spanning phase planes for TS records 1 and 2.

We will use the term **time series (TS) record** to refer to a data file such as a spreadsheet with the first column giving time increasing downwards and additional column giving the values of q_i (a state variable) and $\dot{q}_i$ (time derivative of q_i) for $i = 1, 2, \ldots N$ at the time indicated by the first column. Any q_i or $\dot{q}_i$ is called a **state variable**. The multidimensional space $(q_i)_{i=0}^N$ is called **configuration space** The multidimensional space $(q_i, \dot{q}_i)_{i=0}^N$ is called **phase space**. A two dimensional space $(q_i, \dot{q}_i)$ for any i is called a **phase plane**.

The TS record may contain initially only a column for time and columns for configuration space, if the change in time between rows is small. In that case we can subtract two consecutive q_i readings and divide by the difference in times to obtain the $\dot{q}_i$ column and same row.

For this paper, each q_i represents either a **translation factor (TF)** concentration, or a **gene expression (GE)** concentration.[1]

Suppose x annd y are any two q_i. For $\xi \in \{x \rightarrow y, y \rightarrow x\}$, let
n^ξ_{acc}: number of arrows , initially zero
n^ξ_{rej}: number of arrows rejected, initially zero
$N^\xi = n^\xi_{acc} + n^\xi_{rej}$: number of arrows detected
$p^\xi_{acc} = \frac{n^\xi_{acc}}{n^\xi_{acc}+n^\xi_{rej}}$: probability of causal arrow, initially zero.
N^*: threshold value for N^ξ
p^*_{acc}: threshold value for p^ξ_{acc}

The MM algorithm for finding an AR net from 2 TS records, consists of the following steps:

[1] The terms "translation factor" and "gene expression" are both defined in the AR net chapter of my free book Ref.[2]

1. **Compare 2 TS records and score arrows between any two autoregulons nodes $\boxed{\underline{x}}$ and $\boxed{\underline{y}}$**

   Consider Fig.1. In that figure, suppose the two ends of bridge a are equal: $\mathcal{X}_1(t_a) \approx \mathcal{X}_2(t'_a)$ and the two ends of bridge b are equal too: $\mathcal{Y}_1(t_b) \approx \mathcal{Y}_2(t'_b)$. [2]

   At the very least, one must store (unless they are the default value zero) the current values of n^{ξ}_{acc} and n^{ξ}_{rec} for $\xi \in \{x \to y, y \to x\} = \mathcal{A}$, where x and y are any two q_i.

   - if $t_a < t_b$ and $t'_a < t'_b$ (bridges are parallel in time)[3]

   $$\begin{cases} n^{x\to y}_{acc} + + \\ N^{x\to y} + + \end{cases} \tag{1}$$

   - if $t_a > t_b$ and $t'_a > t'_b$ (bridges are parallel in time)

   $$\begin{cases} n^{y\to x}_{acc} + + \\ N^{y\to x} + + \end{cases} \tag{2}$$

   - if $t_a < t_b$ and $t'_a > t'_b$ (bridges are crossing in time)

   $$\begin{cases} n^{x\to y}_{rej} + + \\ N^{x\to y} + + \end{cases} \tag{3}$$

   - if $t_a > t_b$ and $t'_a < t'_b$ (bridges are crossing in time)

   $$\begin{cases} n^{y\to x}_{rej} + + \\ N^{y\to x} + + \end{cases} \tag{4}$$

2. **Draw DAG**

   $$\begin{tikzcd} \boxed{\underline{x}} \arrow[r, bend right, "p^{x\to y}_{acc}(N^{x\to y})"'] & \boxed{\underline{y}} \arrow[l, bend right, "p^{y\to x}_{acc}(N^{y\to x})"'] \end{tikzcd} \tag{5}$$

   If x and y are any two q_i, draw an arrow from autoregulon $\boxed{\underline{x}}$ to autoregulon $\boxed{\underline{x}}$ iff both $p^{x\to y}_{acc} > p^*$ and $N^{x\to y} > N^*$ are true.

   Likewise, draw an arrow from autoregulon $\boxed{\underline{y}}$ to autoregulon $\boxed{\underline{x}}$ iff both $p^{y\to x}_{acc} > p^*$ and $N^{y\to x} > N^*$ are true.

   When drawing an arrow, put the values p^{ξ}_{acc} and N^{ξ} over the arrow, where $\xi \in \mathcal{A}$. See Eq.(5) where this is done with variables. Do it with the values of those variables instead.

[2] By $\mathcal{X} \approx \mathcal{Y}$ we mean that both of these vectors are inside the same small bin or open ball of size given by a pre-specified precision.

[3] $x + +$ means add 1 to x

At first glance, Eq.(5) doesn't look like a DAG, because it has a cycle and DAGs are, by definition, acyclic. But Eq.(5) does indeed represent a DAG because, as explained in the AR net chapter of Ref.[2], Eq.(5) represents this net:

$$\begin{tikzcd} \underline{x} \arrow[d] \arrow[dr] & \underline{y} \arrow[d] \arrow[dl] \\ \underline{\dot{x}} & \underline{\dot{y}} \end{tikzcd} \tag{6}$$

which is acyclic.

If R_1 and R_2 are two TS records and G is the DAG obtained by following the MM algorithm presented above, then one can represent that TS algorithm diagrammatically by

$$\begin{tikzcd} R_1 \arrow[dr] & R_2 \arrow[d] \\ & G \end{tikzcd} \tag{7}$$

In the next section, we will try to define the merging of more than two records into a single DAG.

3 Comparing More Than 2 TS Records

3.1 Merging two or more DAGs into one DAG

- merging doubly overlapping arrows (overlap in head and tail)

$$\left.\begin{array}{c} \boxed{\underline{x}} \xrightarrow{p_1(N_1)} \boxed{\underline{y}} \\ \\ \boxed{\underline{x}} \xrightarrow{p_1(N_2)} \boxed{\underline{y}} \end{array}\right\} \Longrightarrow \boxed{\underline{x}} \xrightarrow{p(N)} \boxed{\underline{y}} \tag{8}$$

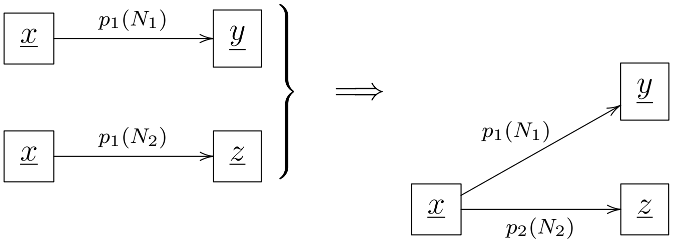
where

$$N = N_1 + N_2 \tag{9}$$

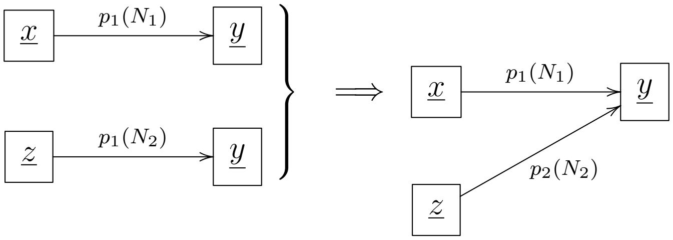
and

$$p = \frac{\sum_{i=1}^{2} n_{acc,i}}{\sum_{i=1}^{2}(n_{acc,i} + n_{rej,i})} = \frac{p_1 N_1 + p_2 N_2}{N_1 + N_2} = p_1 \frac{N_1}{N} + p_2 \frac{N_2}{N} \tag{10}$$

- merging singly overlapping arrows (overlap in head or tail but not both)

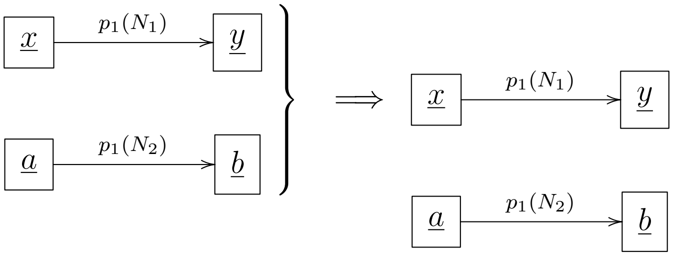


(11)



(12)

- non-overlapping arrows

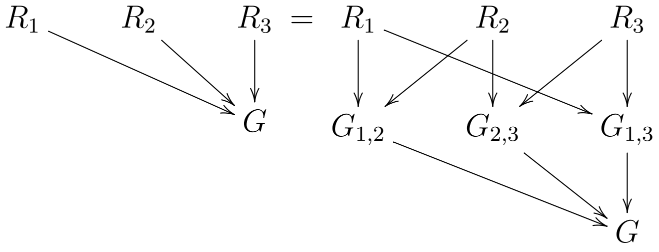


(13)

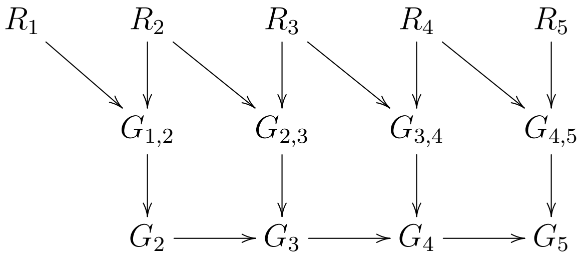
If G_1 and G_2 are two DAGs and G is the DAG obtained by following the DAG merging lgorithm presented above, then one can represent that DAG merging algorithm diagrammatically by

$$\begin{tikzcd}
G_1 \arrow[dr] & G_2 \arrow[d] \\
& G
\end{tikzcd} \tag{14}$$

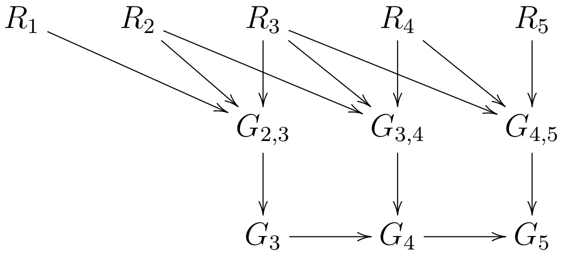
$$\begin{tikzcd}
G_1 \arrow[drr] & G_2 \arrow[dr] & G_3 \arrow[d] & = & G_1 \arrow[dr] & G_2 \arrow[d] & G_3 \arrow[dd] \\
& & G & & & G_{1,2} \arrow[dr] & \\
& & & & & & G
\end{tikzcd} \tag{15}$$



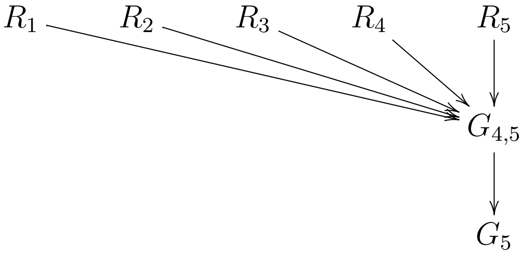
(16)



(17)



(18)



(19)

References

[1] Uri Alon. *An introduction to systems biology: design principles of biological circuits*. Chapman and Hall/CRC, 2019.

[2] Robert R. Tucci. Bayesuvius (free book). `https://github.com/rrtucci/Bayesuvius`.

[3] Robert R. Tucci. https://github.com/rrtucci/causalfitbit (software and paper). `https://github.com/rrtucci/CausalFitbit`.

[4] Robert R. Tucci. Mappa Mundi (software and paper). `https://github.com/rrtucci/mappa_mundi`.