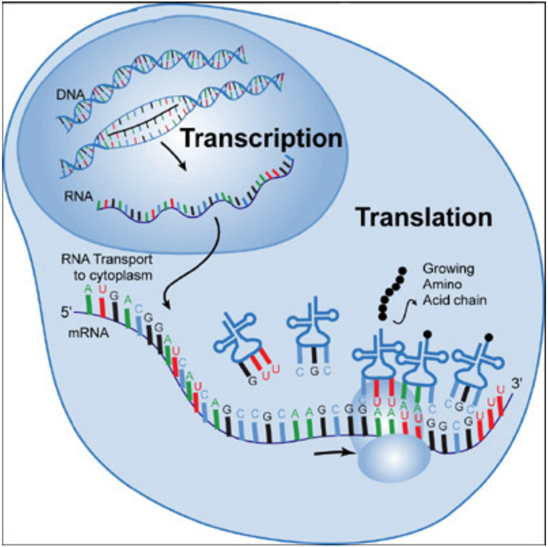
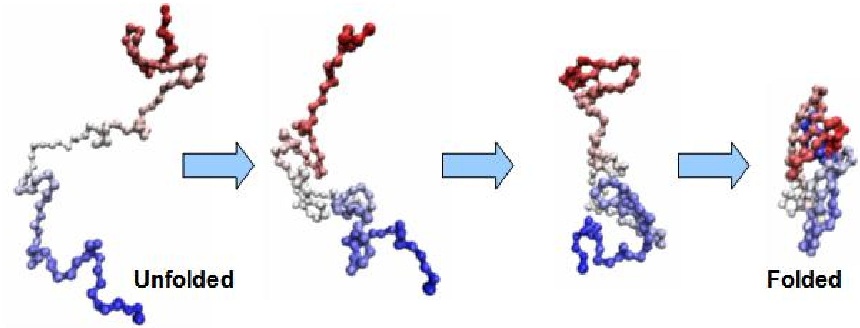
Models and algorithms for deciphering transcriptional regulations coordinating the response to environmental stresses

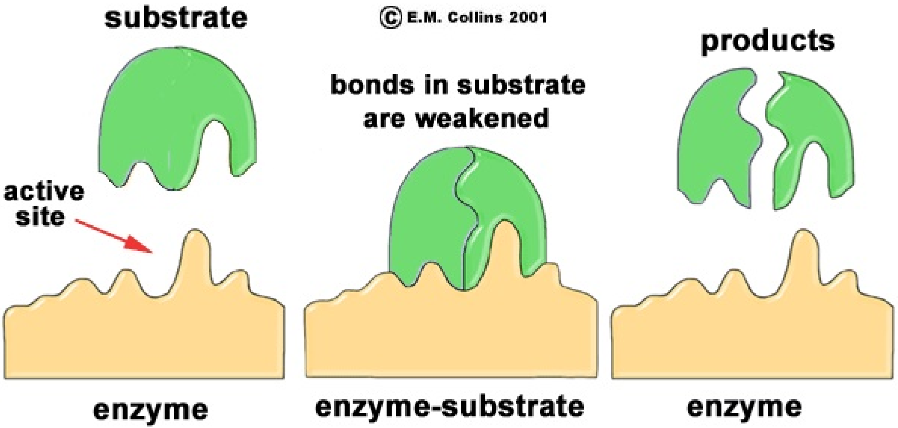
Vicente Acuña Aguayo

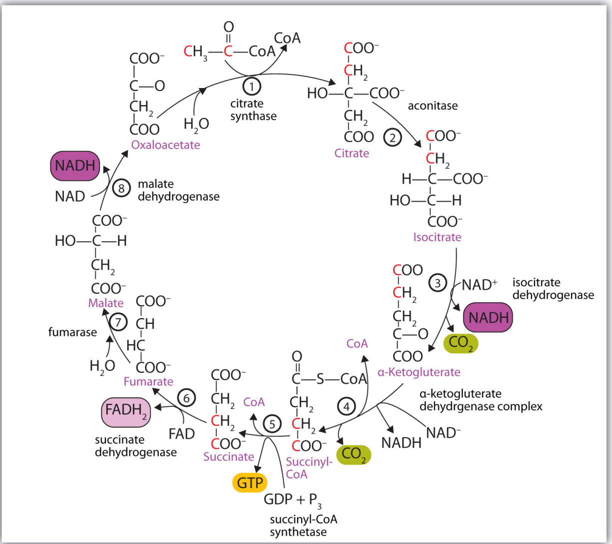
Scientist
Centro de Modelamiento Matemático (CMM),
Universidad de Chile

Workshop :
Desafíos matemáticos e informáticos
para la construcción y análisis de redes de regulación biológica
U. de Concepción 22-23 Abril 2016


DNA
Transcription
RNA
Translation
RNA Transport
to cytoplasm
Growing
Amino
Acid chain
5'
mRNA
3'


Unfolded
Folded


substrate
© E.M. Collins 2001
products
bonds in substrate
are weakened
active
site
enzyme
enzyme-substrate
enzyme


citrate synthase
aconitase
isocitrate dehydrogenase
α-ketogluterate dehydrgenase complex
succinyl-CoA synthetase
succinate dehydrogenase
fumarase
malate dehydrogenase
Oxaloacetate
Citrate
Isocitrate
α-Ketogluterate
Succinyl-CoA
Succinate
Fumarate
Malate
NADH
NAD
NAD+
NAD-
CO2
CoA
GTP
GDP + P3
FADH2
FAD
H2O
1
2
3
4
5
6
7
8

Levels of regulation in bacterial gene expression

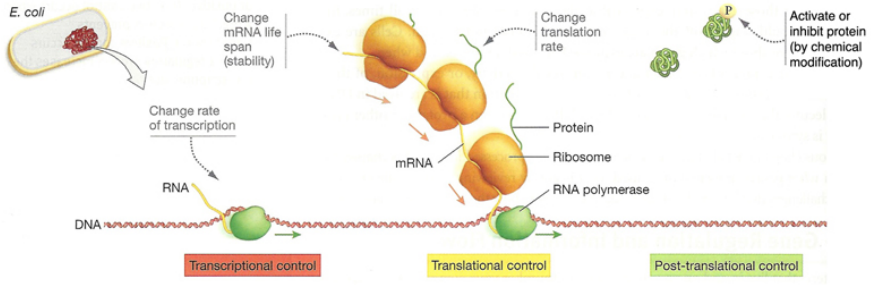
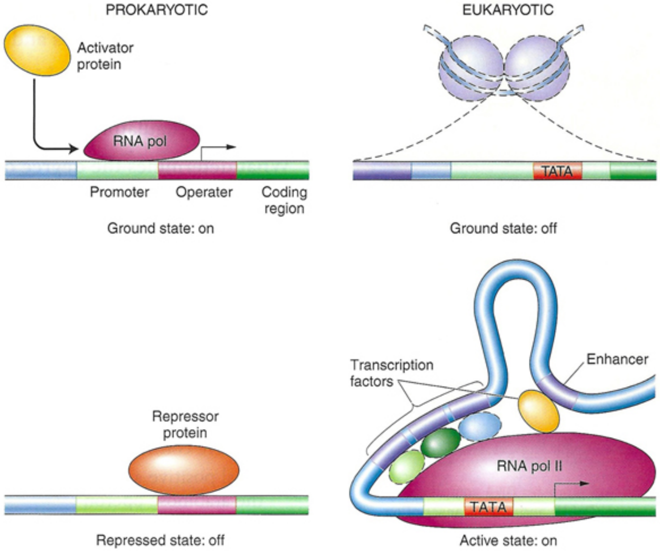


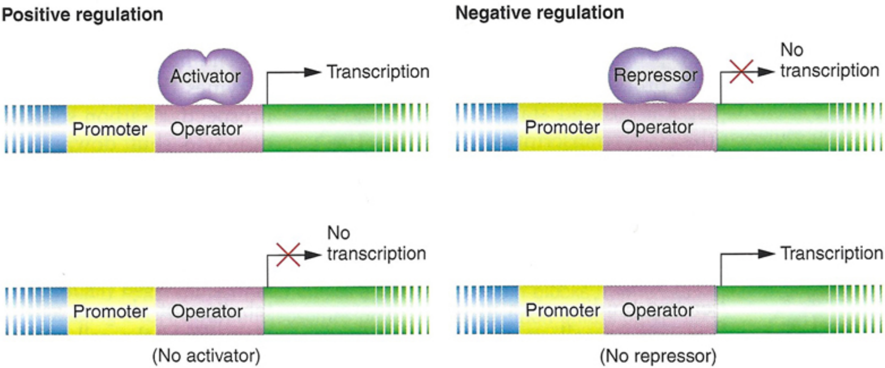
FIGURE 17.1 Gene Expression in Bacteria Can Be Regulated at Three Levels.

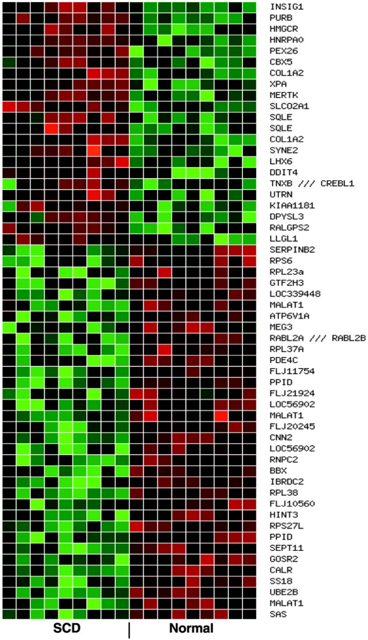
✔**EXERCISE** Label the mode of regulation that is the slowest in response time, and that which is fastest. Label the most efficient and least efficient in resource use.

Overview of transcriptional regulation



Positive and negative regulation




INSIG1
PURB
HMGCR
HNRPA0
PEX26
CBX5
COL1A2
XPA
MERTK
SLCO2A1
SQLE
SQLE
COL1A2
SYNE2
LHX6
DDIT4
TNXB /// CREBL1
UTRN
KIAA1181
DPYSL3
RALGPS2
LLGL1
SERPINB2
RPS6
RPL23a
GTF2H3
LOC339448
MALAT1
ATP6V1A
MEG3
RABL2A /// RABL2B
RPL37A
PDE4C
FLJ11754
PPID
FLJ21924
LOC56902
MALAT1
FLJ20245
CNN2
LOC56902
RNPC2
BBX
IBRDC2
RPL38
FLJ10560
HINT3
RPS27L
PPID
SEPT11
GOSR2
CALR
SS18
UBE2B
MALAT1
SAS
SCD
Normal

Input Data

Let $\mathbb{G}$ be the set of genes in the studied organism. The method requires:

- *Affinity pairs:* A set $\mathcal{A} \subseteq \mathbb{G} \times \mathbb{G}$ of gene pairs obtained by TF/BS sequence affinity and the *p*-values associated. This forms a directed graph $\mathcal{G}$.

Input Data

Let $\mathbb{G}$ be the set of genes in the studied organism. The method requires:

- *Affinity pairs:* A set $\mathcal{A} \subseteq \mathbb{G} \times \mathbb{G}$ of gene pairs obtained by TF/BS sequence affinity and the *p*-values associated. This forms a directed graph $\mathcal{G}$.
- *Co-expressed pairs:* A set $\mathcal{C} \subseteq \mathbb{G} \times \mathbb{G}$ of pairs of co-expressed genes.

Input Data

Let $\mathbb{G}$ be the set of genes in the studied organism. The method requires:

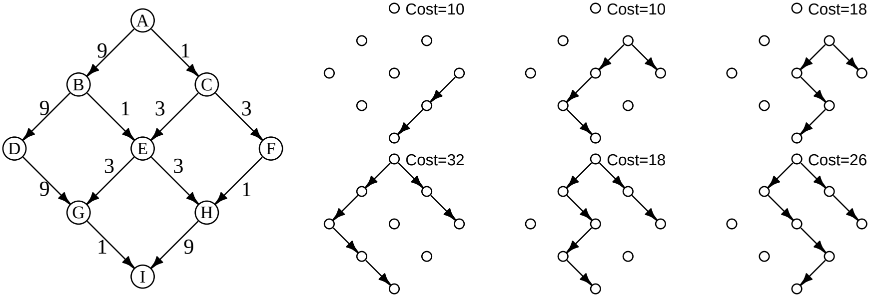
- *Affinity pairs:* A set $\mathcal{A} \subseteq \mathbb{G} \times \mathbb{G}$ of gene pairs obtained by TF/BS sequence affinity and the p-values associated. This forms a directed graph $\mathcal{G}$.
- *Co-expressed pairs:* A set $\mathcal{C} \subseteq \mathbb{G} \times \mathbb{G}$ of pairs of co-expressed genes.
- *Validated pairs:* Optionally, $\mathcal{G}$ can also include a set $\mathcal{V} \subseteq \mathbb{G} \times \mathbb{G}$ of gene pairs corresponding to independent experimentally validated regulations, if available.

Definition

Given a pair $(A, B) \in \mathcal{C}$ of co-expressed genes, an *explanation for* (A, B) in $\mathcal{G}$ is a set of arcs $\mathcal{E}$ that satisfy any of the following conditions:

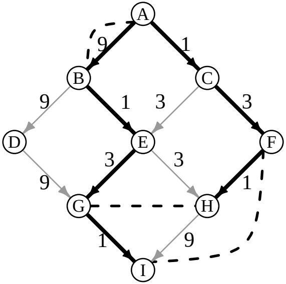
- $\mathcal{E}$ is a directed path from A to B;
- $\mathcal{E}$ is a directed path from B to A;
- $\mathcal{E}$ is the union of two divergent directed paths starting from some gene C and arriving respectively at A and B, having only vertex C in common.

- But indeed, we don't want to recover any explanation for each pair, but just to foster simple (short) and confident explanations.
- For each TF/BS affinity, we define a cost on each arc depending on the *p*-value associated to the affinity.
- The cost of an explanation is the sum of the cost of its arcs.



All explanations for the co-expression $\{F, I\} \in \mathcal{C}$ and the cost associated.

We say that a subgraph $\mathcal{G}' \subseteq \mathcal{G}$ *explains* $\mathcal{C}$ if for every pair $(A, B) \in \mathcal{C}$ the subgraph $\mathcal{G}'$ contains an explanation for (A, B).



In the figure: three subgraphs explaining $\mathcal{C} = \{\{A, B\}, \{F, I\}, \{G, H\}\}$.
Only the first and the last are minimal ones.

Modelling objective: to define a *good* subgraph explaining all co-regulations in $\mathcal{C}$.

Algorithmic objective: to enumerate all those subgraphs.

How difficult is enumerate all minimal subgraphs (without considering costs)?

Problem

$\textsc{EnumCohe}(\mathcal{G}, \mathcal{C})$: Given an oriented graph $\mathcal{G}$ and a set of pairs of vertices $\mathcal{C}$, enumerate all minimal subgraphs of $\mathcal{G}$ that explain $\mathcal{C}$.

How difficult is enumerate all minimal subgraphs (without considering costs)?

Problem

$\textsc{EnumCohe}(\mathcal{G}, \mathcal{C})$: Given an oriented graph $\mathcal{G}$ and a set of pairs of vertices $\mathcal{C}$, enumerate all minimal subgraphs of $\mathcal{G}$ that explain $\mathcal{C}$.

- $\textsc{EnumCohe}$ is hard: enumerate all minimal subgraphs that explain $\mathcal{C}$ cannot be done in polynomial total time unless $P = NP$ (we reduce from *path conjunction problem*).
- We can still try to develop some heuristics. . .
- but indeed this model is not very interesting: it generates too many solutions.

More interesting:

Problem

$\textsc{EnumMinCohe}(\mathcal{G}, \mathcal{C})$: Given an oriented graph $\mathcal{G}$ and a set of pairs of vertices $\mathcal{C}$, enumerate all subgraph explaining $\mathcal{C}$ of minimum total cost.

More interesting:

Problem

$\textsc{EnumMinCohe}(\mathcal{G}, \mathcal{C})$: Given an oriented graph $\mathcal{G}$ and a set of pairs of vertices $\mathcal{C}$, enumerate all subgraph explaining $\mathcal{C}$ of minimum total cost.

- Unfortunately, even finding **one** subgraph of minimum cost is *NP*-hard (reduction from Steiner Weighted Directed Tree problem).
- Again, we can use some heuristics...
- but indeed this model is not very robust: adding a new pair in $\mathcal{C}$ can (in theory) change dramatically the set of solutions.

Maybe the cell (or evolution) uses a “local parsimony”:

Maybe the cell (or evolution) uses a “local parsimony”:

Definition

We say that an explanation $\mathcal{E}$ for the pair (A, B) in $\mathcal{C}$ is *optimal* if it is of minimum cost among all the explanations for the pair.

Maybe the cell (or evolution) uses a "local parsimony":

Definition

We say that an explanation $\mathcal{E}$ for the pair (A, B) in $\mathcal{C}$ is *optimal* if it is of minimum cost among all the explanations for the pair.

Now, we can define the solutions that contain an optimal explanation for every pair in $\mathcal{C}$.

Definition

We say that a subgraph is an *optimal* subgraph explaining $\mathcal{C}$ if it is the union of $|\mathcal{C}|$ optimal explanations, one for each pair $(A, B) \in \mathcal{C}$.

Definition

We say that a subgraph is an *optimal* subgraph explaining $\mathcal{C}$ if it is the union of $|\mathcal{C}|$ optimal explanations, one for each pair $(A, B) \in \mathcal{C}$.

- Instead of enumerating all *optimal* subgraphs, we define $\mathcal{G}_L$ the union of all *optimal* subgraph explaining $\mathcal{C}$.
- To compute $\mathcal{G}_L$, we need simply to compute all optimal explanations for each pair in $\mathcal{C}$.
- The software LOMBARDE uses an algorithm of minimum paths in modified graph to compute $\mathcal{G}_L$.

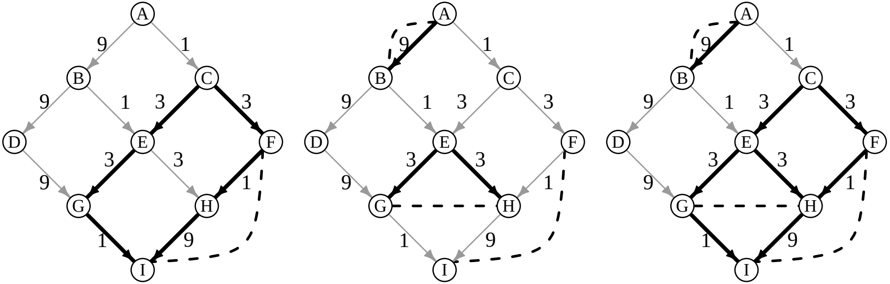
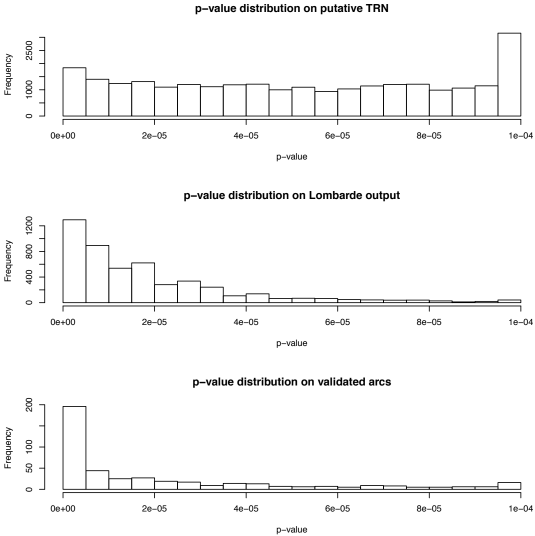
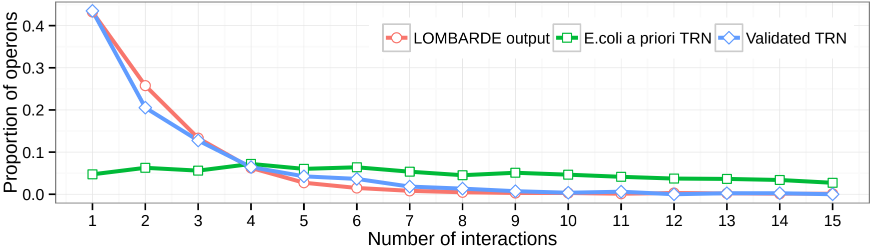

A
9 1
B C
9 1 3 3
D E F
3 3
9 1
G H
1 9
I
A
9 1
B C
9 1 3 3
D E F
3 3
9 1
G H
1 9
I
A
9 1
B C
9 1 3 3
D E F
3 3
9 1
G H
1 9
I

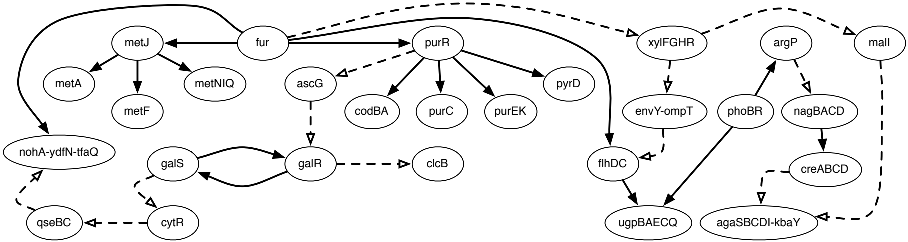
Table 1 Characteristics of the *a priori* graphs and LOMBARDE output networks

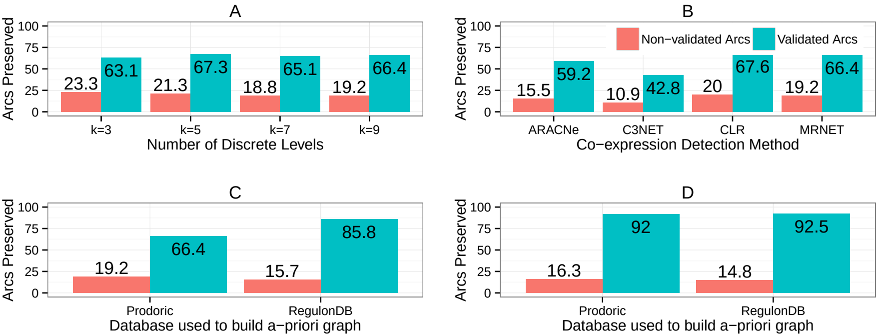
Network	Explained co-expressions		No . of vertices	No. of arcs	No. of arcs in $\mathcal{V}$
TRN $\mathcal{G}_{\mathcal{V}}$ built from $\mathcal{V}$	3,990	(6.5 %)	823	1,652	1,652
E. coli ab initio $\mathcal{G}_{\mathcal{A}}$	56,044	(91.1 %)	2,390	25,604	444
LOMBARDE output for $\mathcal{G}_{\mathcal{A}}$	56,044	(91.1 %)	2,336	4,922	295
E. coli extended $\mathcal{G}_{\mathcal{AV}}$	56,789	(92.3 %)	2,434	26,812	1,652
LOMBARDE output for $\mathcal{G}_{\mathcal{AV}}$	56,789	(92.3 %)	2,370	4,374	1,520

The a priori graphs explained most of the co-expressions. The LOMBARDE results kept most of the vertices, significantly reduced the number of arcs, and kept most of the validated arcs


p−value distribution on putative TRN
Frequency
0
1000
2500
0e+00
2e−05
4e−05
6e−05
8e−05
1e−04
p−value
p−value distribution on Lombarde output
Frequency
0
400
800
1200
0e+00
2e−05
4e−05
6e−05
8e−05
1e−04
p−value
p−value distribution on validated arcs
Frequency
0
50
100
200
0e+00
2e−05
4e−05
6e−05
8e−05
1e−04
p−value


Proportion of operons
0.0
0.1
0.2
0.3
0.4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
Number of interactions
LOMBARDE output
E.coli a priori TRN
Validated TRN


fur
metJ
purR
xylFGHR
argP
malI
metA
metF
metNIQ
ascG
codBA
purC
purEK
pyrD
envY-ompT
phoBR
nagBACD
nohA-ydfN-tfaQ
galS
galR
clcB
flhDC
creABCD
qseBC
cytR
ugpBAECQ
agaSBCDI-kbaY



A
Arcs Preserved
100
75
50
25
0
23.3
63.1
21.3
67.3
18.8
65.1
19.2
66.4
k=3
k=5
k=7
k=9
Number of Discrete Levels
B
Non−validated Arcs
Validated Arcs
15.5
59.2
10.9
42.8
20
67.6
19.2
66.4
ARACNe
C3NET
CLR
MRNET
Co−expression Detection Method
C
19.2
66.4
15.7
85.8
Prodoric
RegulonDB
Database used to build a−priori graph
D
16.3
92
14.8
92.5
Prodoric
RegulonDB
Database used to build a−priori graph