An Introduction to Reversible Jump MCMC for Bayesian Networks, with Application

Stephen Jensen, CleverSet, Inc.

The research described in this presentation has been funded by the U.S. Environmental Protection Agency through the STAR Cooperative Agreement #CR82-9095 to Colorado State University. It has not been subjected to the Agency's review and therefore does not necessarily reflect the views of the Agency, and no official endorsement should be inferred.

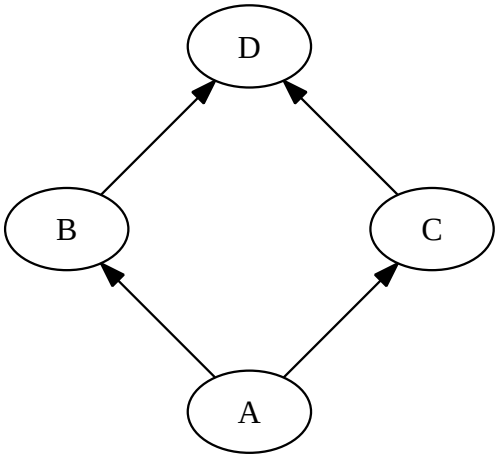
Introduction

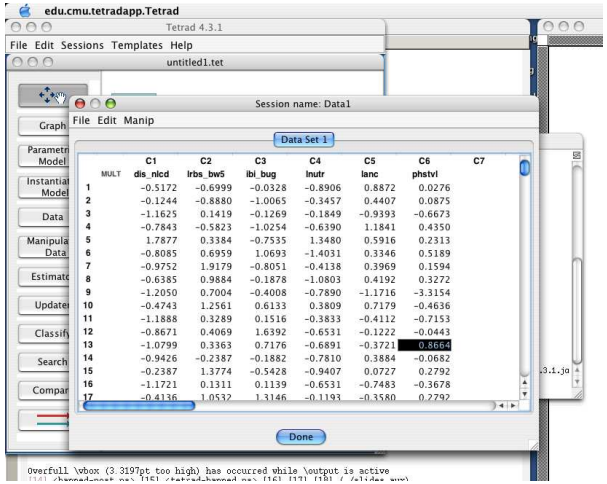
Bayesian Networks

- Bayesian Networks (BNs) are a method for representing complex multivariate probability distributions.
  - Graphical Structure: Directed Acyclic Graph
    - Each **node** represents a variable
    - An **edge** represents an association between two variables
  - Parametrical Model: Multivariate Gaussian
    - Data are independent, multivariate Gaussian
    - Parameters are multivariate Gaussian
    - Marginal distribution of each node (variable) is a multiple regression

Introduction

Bayesian Networks





- **Important point:** despite the arrows, BNs represent a probability distribution and do not imply causations, only associations.

Introduction

Reversible Jump Markov chain Monte Carlo

- *MCMC* is a method for simulating a probability distribution that cannot be directly simulated. MCMC is often used for model selection, especially in very large model spaces.
- *RJMCMC* is a type of MCMC that allows for dimensional changes in the probability distribution being simulated. RJMCMC can be used for model selection in cases where dimensionality may change, such as:
  - ARIMA time series models
  - Gaussian Mixtures

RJMCMC for BNs

- RJMCMC is suited to searching for BNs because:
  - A change in the graphical structure of a BN results in a change in the number of parameters
  - The number of possible structures of BNs increases super-exponentially in the number of variables.

RJMCMC for BNs

- How it works:
  - RJMCMC randomly “walks around” the space of possible model structures by changing one edge at a time – called structural learning.
  - At each step in its “walk”, all of the model parameters are updated – called parametrical learning.
  - At the end, you have a list of all of the model structures it visited at each step and their corresponding set of parameters.

Example: MAHA-MAIA

- Subset of numerical data taken from the MAHA-MAIA study.
- Six variables chosen, with help of a domain expert:
  - Insect IBI - Insect Index of Biotic Integrity
  - Sediment Disturbance - A log-scale metric of excess grain-size diameter
  - Environmental disturbance - Combined percentages of disturbance (urban, agricultural, and mining disturbances)
  - pH
  - Natural logarithm of nutrients (maximum of either N or P)
  - Natural logarithm of slightly translated Acid Neutralizing Capacity

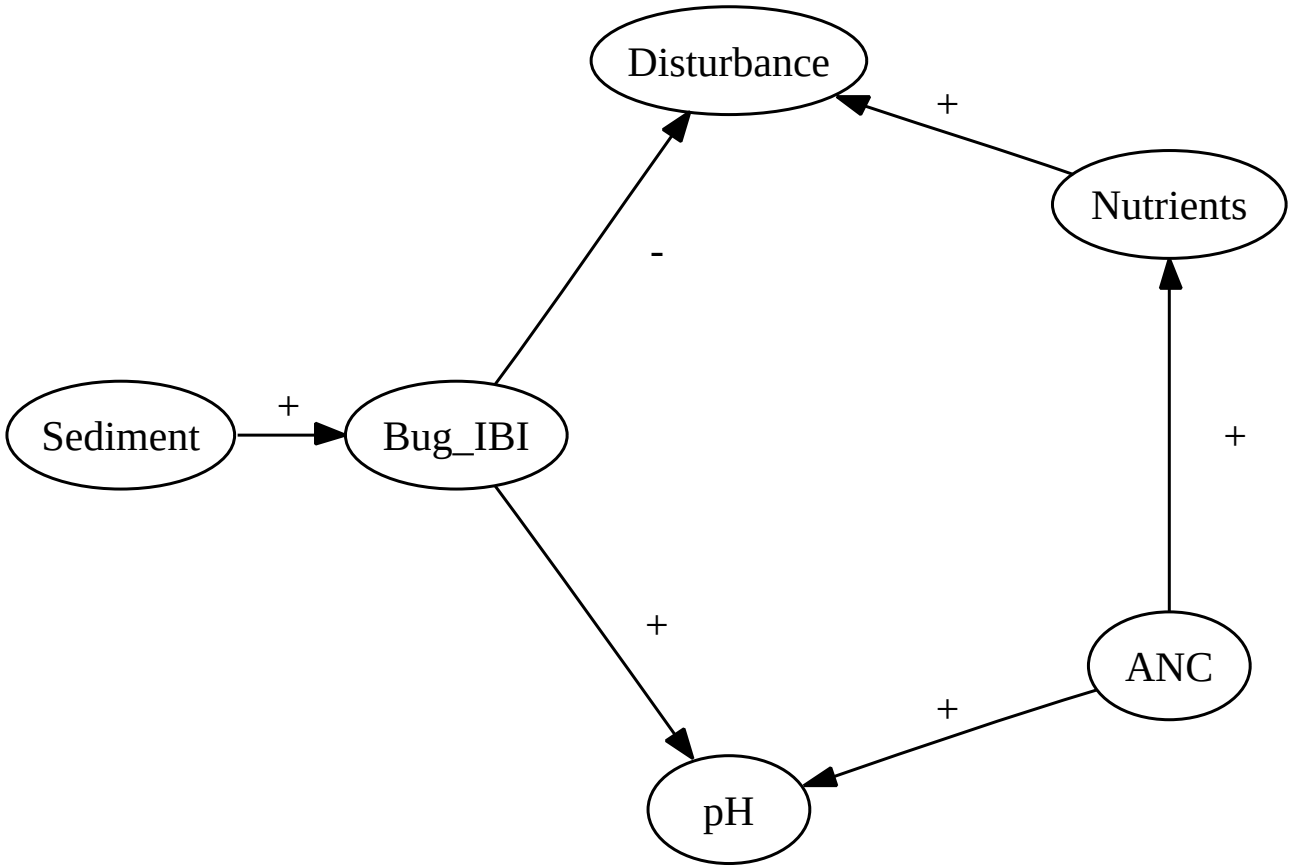
Example: MAHA-MAIA

Implementation notes

- To avoid numerical issues, variables were standardized.
- Three graphical structures are presented:
  - The structure with the *maximum posterior probability*,
  - The “average” model, which gives us a sense of the likelihood of an association between pairs of variables,
  - A reference model obtained from the package TETRAD IV.
- +’s and -’s, derived from the learned parameters, denote the type of quantitative association between variables.

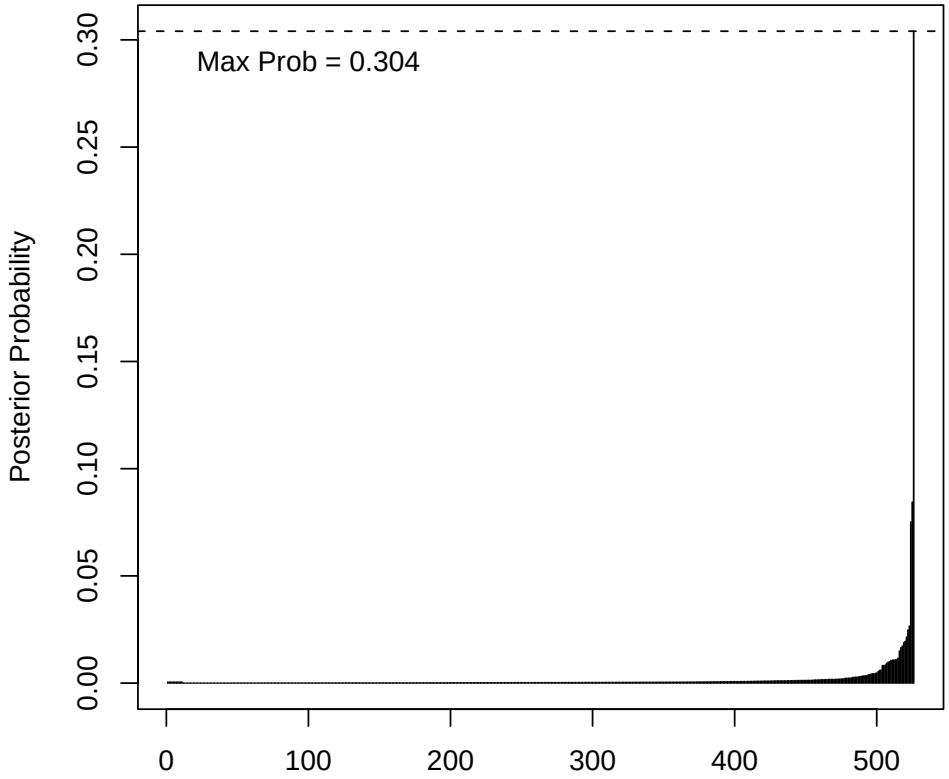
Example: MAHA-MAIA

Maximum posterior probability structure



Example: MAHA-MAIA

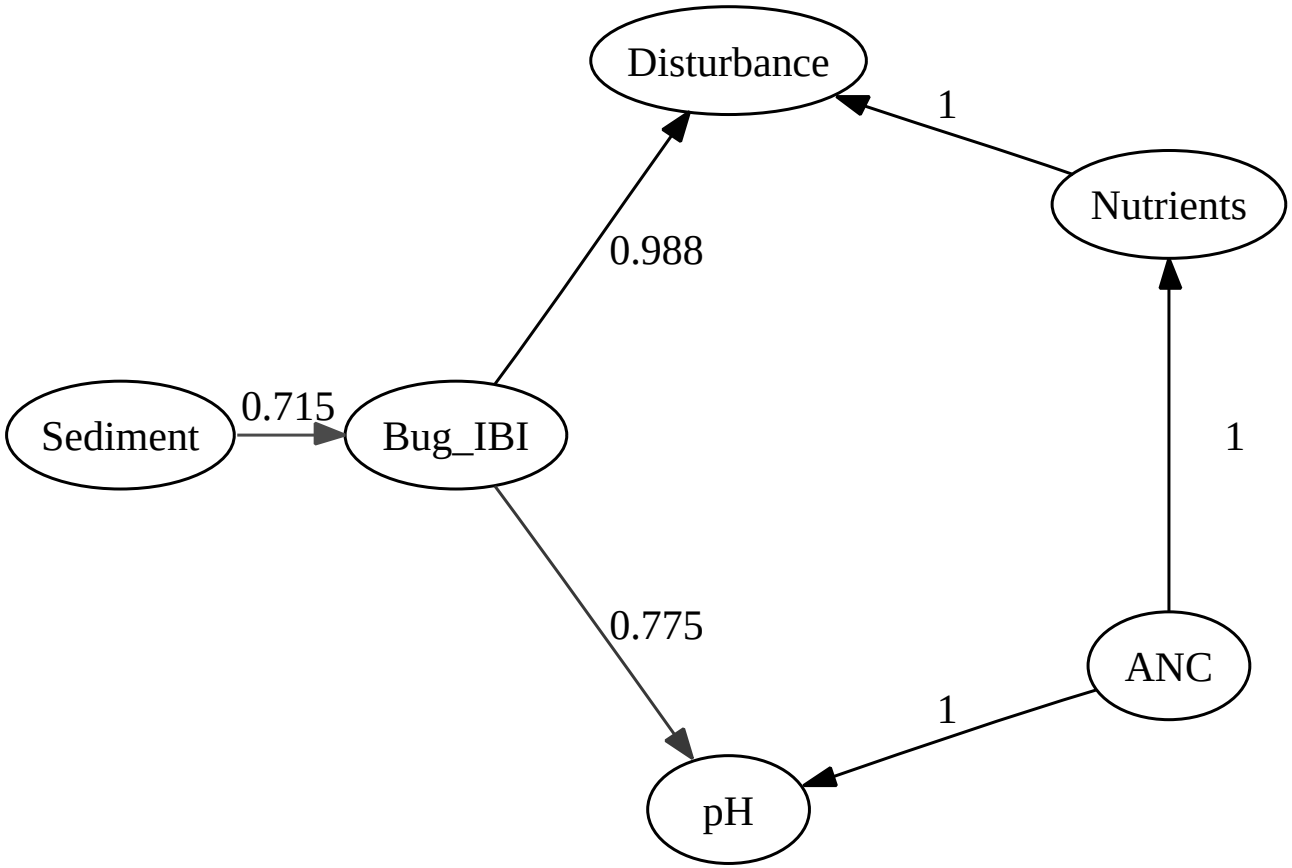
Posterior probabilities of all visited structures



Example: MAHA-MAIA

"Average" structure

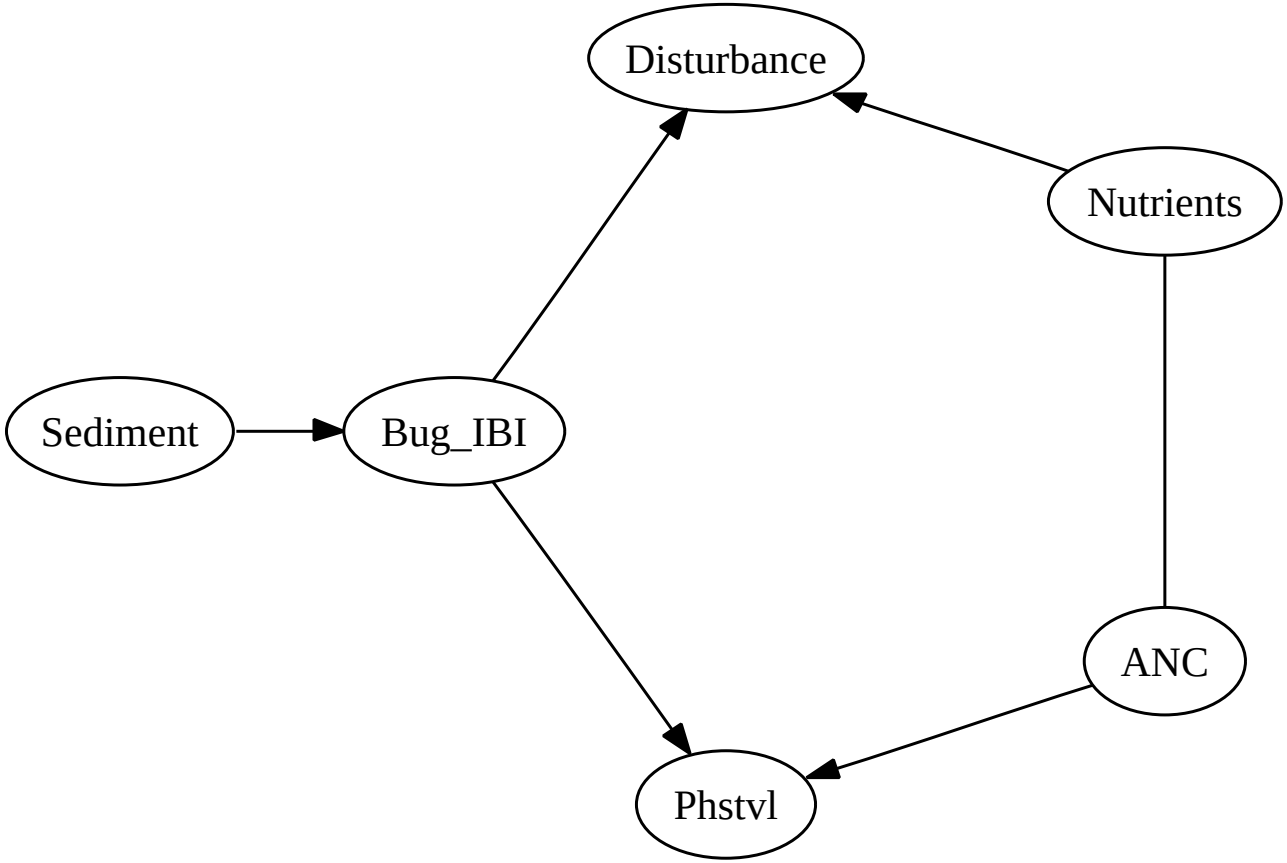
Note: edges with posterior probability $< .3$ *not shown*



Example: MAHA-MAIA

Structure discovered by TETRAD IV

Note: undirected edge on right can be oriented either direction



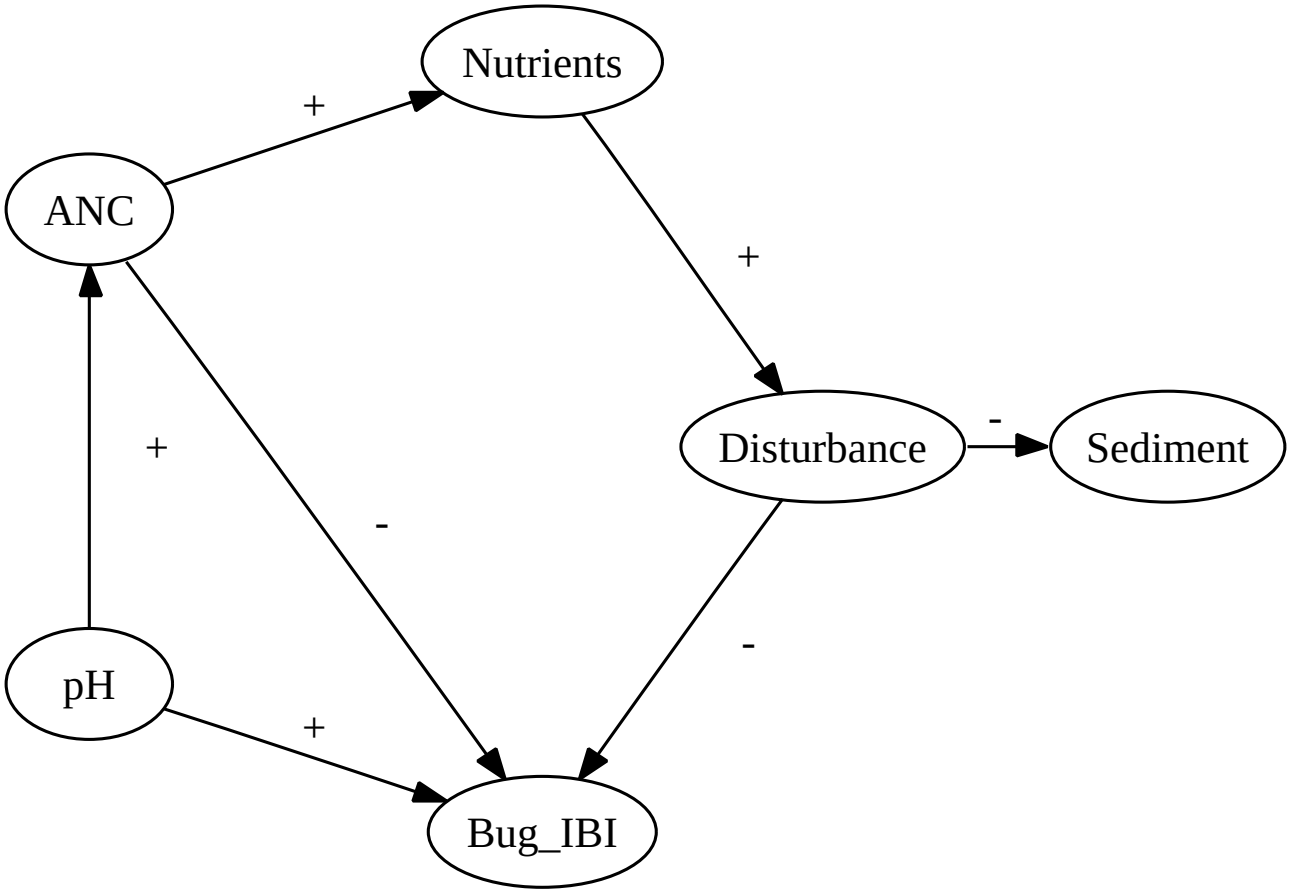
Example: MAHA-MAIA

Discussion

- The +'s and -'s seem reasonable, and the structures agree, but the direction of the edges from Bug_IBI seem to preclude a causal interpretation.
- Recall that BNs encode a joint probability distribution and don't necessarily suggest a causal relation.
- Try again, but remove models from consideration that include edges emanating from Bug_IBI.

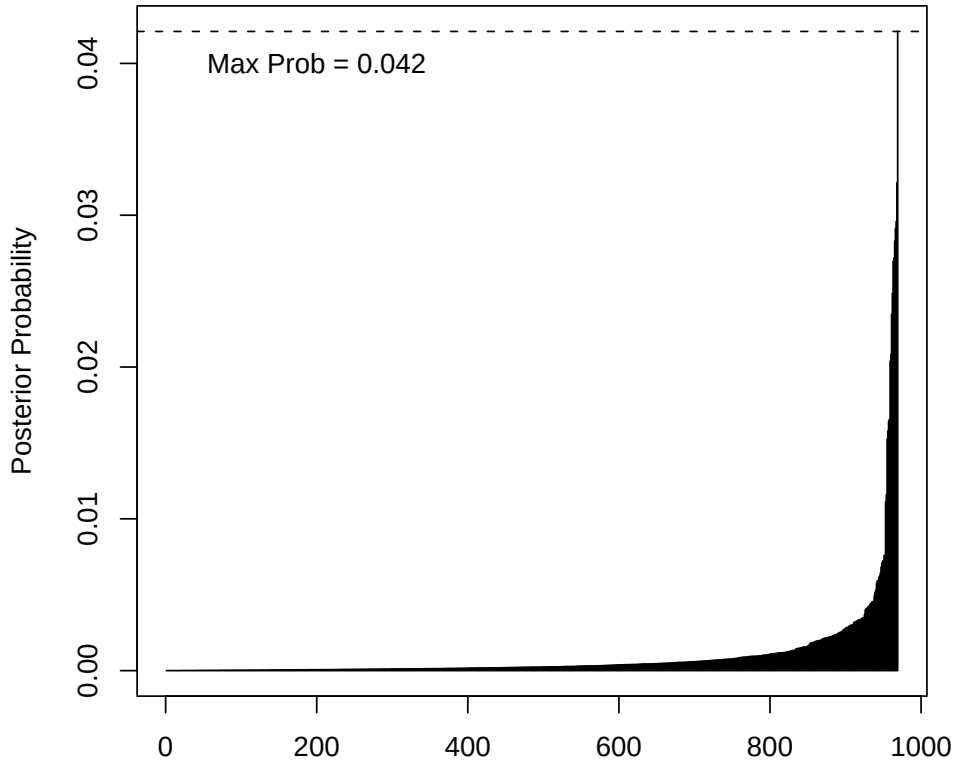
Example: MAHA-MAIA (2)

Maximum posterior probability structure



Example: MAHA-MAIA (2)

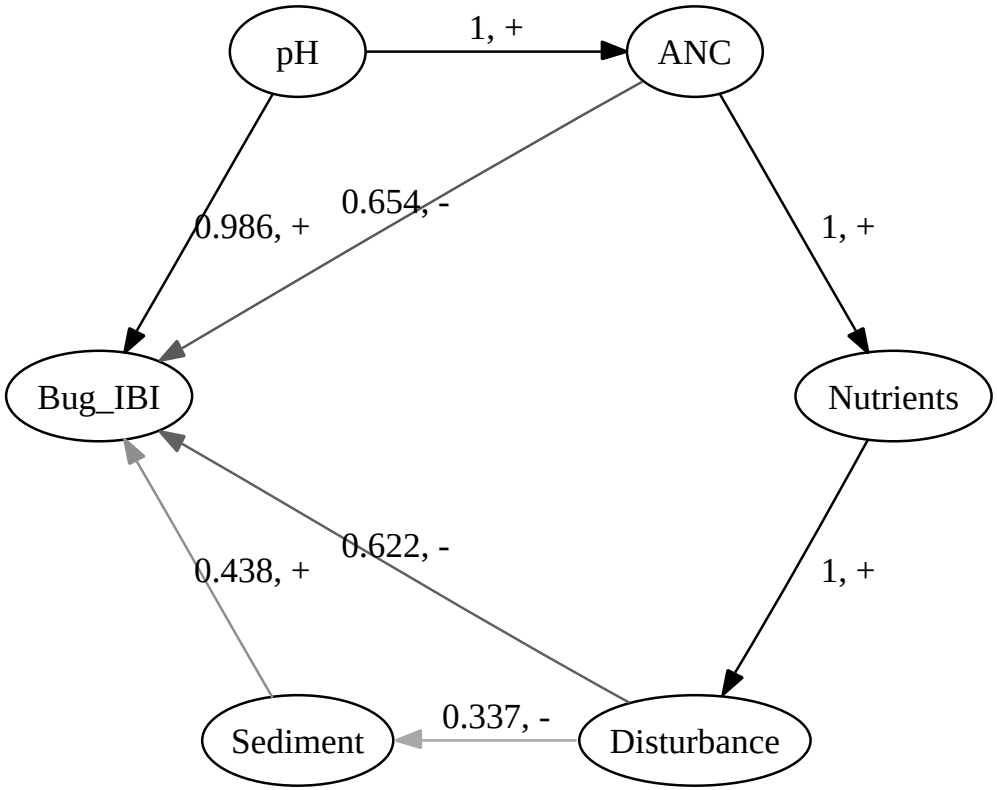
Posterior probabilities of all visited structures



Example: MAHA-MAIA (2)

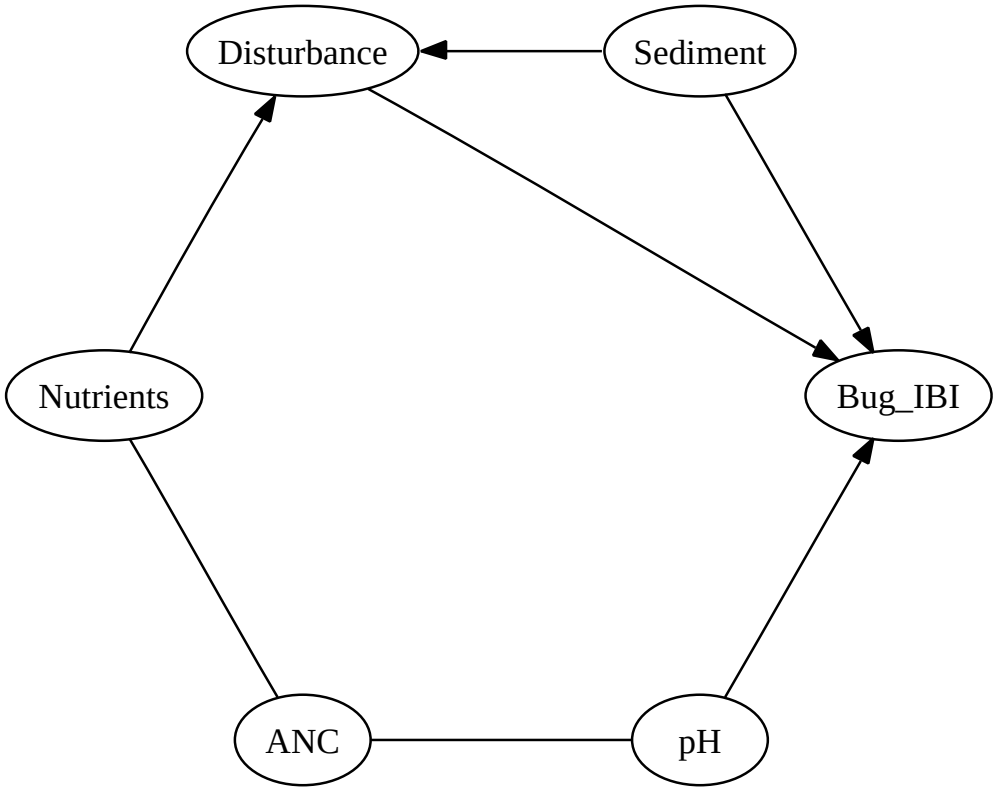
“Average” structure

Note: edges with posterior probability $< .3$ not shown



Example: MAHA-MAIA (2)

Structure discovered by TETRAD IV



Conclusion

- Finding an optimal BN that fits data is known to be a very hard problem (NP-hard, in fact), making heuristic algorithms a necessity.
- Though many of these algorithms are much faster than RJMCMC, increases in computing power and programming refinements are making its lack of speed less of an issue.
- Furthermore, other methods lack the unique strengths of RJMCMC:
  - Posterior edge probabilities give a measure of the likelihood of association
  - Combined structure discovery and parameter learning
  - Fully Bayesian solution