Outline BMC paper

'Module networks' has emerged as (one of?) the most successful method(s) for inferring regulatory networks from microarray data alone. We use synthetic data generated by SynTReN to carry out an extensive test of how well an optimal module network can reproduce the original network. So far module networks have only been tested in two ways:

- On synthetic data generated by a known module network: this is not realistic because a true biological network will never be exactly of the module network form and will therefore be much harder to infer.
- On real data: this is rather limiting because the true underlying network is only known very partially.

With SynTReN we can generate realistic simulated microarray data and answer questions such as:

- What are sensitivity, specificity, etc. of the inferred network?
- How do these depend on the size of the network? On the number of experiments?
- Can the Bayesian score be used to determine parameter settings that give maximal sensitivity, specificity?
- Can the true positives be separated from the false positives by objective measures such as
  - the location of the regulator in the regulation program (top or bottom of the tree)?
  - the Bayesian score of the module it regulates?
  - the regulator assignment entropy?
- . . . ?

We have developed our own software to find an optimal module network because:

- We want to propose improvements/additions to the genomica algorithm:
  - Assignment of regulator/split value pairs should take into account possible noise on the regulator data: in genomica a regulator split has to be 'exact' while sometimes a better partitioning of the module can be obtained if some regulator values are put on the 'wrong' side of the split. The whole Bayesian formalism is used because gene expression is assumed to be stochastic, so it is unnatural to restrict to exact regulator splits.

  - There need not always be just one regulator that can explain a split, it is useful to keep track of which other regulators are equally good, if any.
  - To find an optimal partitioning of the experiments in each module, it is not necessary to know the regulators. Therefore each module can be optimized independently and in parallel, and the time consuming task of assigning regulators that preserve acyclicity needs to be done only once, after the module scores have converged to their final value, and not each time a new regulation program is learned. Hence our program is faster and parts of it can be parallellized in a trivial way.
  - Finding an optimal regulation tree is in fact a hierarchical clustering of the experiments in each module. The genomica way is a top-down hierarchical clustering, but it is known that bottom-up hierarchical clustering usually gives better results.

- There is currently no open source software available (implies that we should make ours open source)

We compare both programs on several synthetic data sets and find that ours is at least as good (even slightly better?) than genomica in terms of sensitivity/specificity, and that the additions such as regulator assignment entropy are really useful in discriminating true from false positives.

Finally we run some tests on real data. Here we use the limited amount of known regulatory interactions:

- to test if our conclusions from using synthetic data are also supported by real data;
- to determine if we can further improve on the separation of true from false positives by adding biological knowledge such as GO enrichment of the regulated modules.
- ...?