Prediction of Uber-operons

10/29/2008
What is an uber-operon?

Operons are relatively conserved in prokaryotes; but the contents of operon are seldom strictly conserved.

Trends Biochem Sci.
What is an uber-operon?

What really conserved is a discrete set of functionally related genes which are often located in the same operon in different species.

Peer BorK’s definition: “… although genomic rearrangements cause variation in the immediate neighborhood of a gene, many genes are maintained over evolutionary time within the context of a discrete set of functionally related genes. We call this set of genes that is conserved at a higher level of organization an uber-operon.”  Trends Biochem Sci. 2000 Oct;25(10):474-9.
An scenario of uber-operon evolution

- The break up and merge of operons are non-random rearrangements of genes, instead, they are maintained within the same transcriptional and regulatory context.

A naïve algorithm for uber-operon prediction

- First, a gene and its orthologs are identified for a chosen number of genomes.

- Second, the conserved neighbors of these orthologous genes are determined.

- Third, the orthologs of these newly determined conserved neighbors are determined in all the genomes.

- The conserved neighbors of these additional orthologs are then determined and added to the ‘set’.

- Steps 1–3 are repeated for several iterations until no new conserved neighbors are found and the number of genes converges to a discrete set, which forms an uber-operon.
A naïve algorithm for uber-operon prediction

- This set often converges to a finite number of genes that are functionally related.
- The addition of more genomes to the procedure does not add significantly to the number of genes in the set.
- A total of 45 genes are predicted to form a translation related uber operon.

Uber-operons can be used for functional predictions

- The flagellar uber-operon in four species

---

Uber-operons can be used for functional predictions

- An transporter urber-operon in four species
- Orthologous relationships are needed to find the true urber-operon.

Uber-operon prediction is not trivial

- When gene duplication is prevalent, identification of orthologs become difficulty, so prediction of urber operons become non-trivial.

- A operational definition of Uber-operon: a set of operons in a genome that are functionally related because the orthologs of genes in different operons are located in the same set of operons in some other genomes.

- Linker genes: a pair of genes in a genome that each is in a different operon and but their orthologous genes are in the same operon in a reference genome.
A heuristic algorithm for uber-operon prediction

- The difference between this algorithm and the naïve algorithm is that the orthologous gene relationship is not assumed; instead orthologous relationship is detected simultaneously with uber-operon prediction.

- Given a target genome $G_1$ and a reference genome $G_2$, assume that each gene in $G_1$ has at most one ortholog in $G_2$, and vice versa.

- Intuitively, a uber-operon is modeled as a maximal group of transcriptionally or functionally related operons that are linked through linker genes;

- The uber operon prediction problem using one reference genome is formulated to find operon mappings that maximizes the overall probability for all the mapped gene pairs to be orthologs.
A heuristic algorithm for uber-operon prediction

- Define a bipartite graph $B = (U, V, E)$ for genomes $G_1$ and $G_2$, where $U$ and $V$ are all the operons in $G_1$ and $G_2$, respectively.

- Operons $U_i$ and $V_j$ in $G_1$ and $G_2$ contain $p_i$ and $q_j$ genes, respectively.

$$U = \bigcup_{i=1}^{m} U_i \quad U_i = \{ u_{i,s} / s = 1, 2, \ldots, p_i \}$$

$$V = \bigcup_{i=1}^{m} V_i \quad V_i = \{ v_{j,s} / s = 1, 2, \ldots, q_j \}$$

Homology

$E = w(u_{i,s}, v_{j,t})$
A heuristic algorithm for uber-operon prediction

- A matching of graph $B$ is defined as a subset of the edges $E$, such that no two edges in the subset share a common vertex.

- Intuitively, a matching represents a one-to-one correspondence between genes in subsets of $U$ and $V$.

$$U = \bigcup_{i=1}^{m} U_i \quad U_i = \{u_{i,s} / s = 1,2,\ldots, p_i \}$$

A match $M$, $|M|=14$

$$E = \{w(u_{i,s}, v_{j,t})\}$$

$$V = \bigcup_{i=1}^{m} V_i \quad V_i = \{v_{j,s} / s = 1,2,\ldots, q_j \}$$
A heuristic algorithm for Uber-operon prediction

For any matching $M$ of $B$, define a multigraph (a graph with multiple edges between the nodes):

$$A_M = (O, M)$$

$$O = \{U_i; V_j \mid 1 \leq i \leq m, 1 \leq j \leq n\}$$

The vertex set $O$ is the operons in $G_1$ and $G_2$, and $M$ the edge set.

$$E = w(u_{i,s}, v_{j,t})$$

$$U = \bigcup_{i=1}^{m} U_i$$

$$V = \bigcup_{j=1}^{n} V_j$$
A heuristic algorithm for uber-operon prediction

- Let $c(M)$ be the number of connected components of $A_M$.
- An uber-operon identification problem is defined as to find the maximal matching $M$ of $B$ that maximizes $c(M)$.

\[ A_M = (O, M) \quad |M| = 14, \quad c(M) = 2 \]

\[ U = \bigcup_{i=1}^{m} U_i \]

\[ E = \{w(u_{i,s}, v_{j,t})\} \]

\[ V = \bigcup_{j=1}^{n} V_j \]
A heuristic algorithm for uber-operon prediction

- The uber-operon identification problem, formulated as above, is NP-hard;

- A heuristic algorithm for this problem:
  - First, find non-overlapping individual operon pairs across $U$ and $V$ that give the highest total matching size among all such operon pairs, they are called operon-group pairs.
  - This can be achieved by first finding one pair of operons that has highest matching size between any possible operon pairs across $U$ and $V$; and then remove this pair from $B$ and repeat this procedure on the updated $B$ until no more operon pairs can be found.
  - Then, merge operon pairs into operon-group pairs if such merging can lead to the increase of the overall matching size. This merge operation is repeated until the objective value $c(M)$ cannot be increased any more. The resulting operon groups in $U$ and $V$ are the predicted uber-operons in the two genomes, respectively.
A heuristic algorithm for uber-operon prediction

A schematic show of the algorithm

A heuristic algorithm for uber-operon prediction

- The uber-operons predicted based on different reference genomes may be different, because each reference genome might provide different ‘reference’ information.

- By effectively combining all these predictions, one could
  
  (i) Eliminate accidental false predictions due to various reasons, such as false operon prediction in a particular reference genome;
  
  (ii) Reduce false negative predictions due to the incomplete (reference) information given by any specific reference genome.
An algorithm for Urber-operon prediction

- Define a weighted graph $G$:
  1. each predicted operon in the target genome is represented as a vertex;
  2. two vertices have an edge between them if and only if the two corresponding operons are predicted to be in the same uber-operon by at least one of the $N$ predictions; and
  3. the weight of an edge is defined to be the number of times that the two corresponding operons are in the same uber-operon among all the $N$ predictions.

In general, $G$ consists of a number of highly connected subgraphs, which can be identified by graph clustering algorithms, such as the Markov cluster algorithm (MCL) [http://micans.org/mcl/].
An algorithm for Über-operon prediction

- Integrate the results from predictions using multiple reference genomes: