12 Repeat resolution

This exposition is based on the following sources, which are all recommended reading:

1. Separation of nearly identical repeats in shotgun assemblies using defined nucleotide positions, DNPs, Tammi et al, Bioinformatics 2002
2. Correcting errors in shotgun sequences, Tammi et al, Nucleic acids research, 2003
3. DNPTrapper: an assembly editing tool for finishing and analysis of complex repeat regions, Arner et al, BMC Bioinformatics, 2005
4. Separating repeats in DNA sequence assembly, Kececioglu and Yu, RECOMB 2001

12.1 Problem definition

In many applications in NGS based Bioinformatics, we face the same problem. We are given a collection of reads which have a high degree of sequence similarity, resulting for example to being mapped at a (but not only) specific genomic location.

- During assembly: Reads from repetitive regions of one genome, or similar reads from closely related organisms in metagenomic samples are falsely drawn together.
- In quantitative analysis: Even if a reference genome is given, reads can be mapped to several locations, or vice versa, to a specific location many reads are mapped that do not stem from that region (or even that organism). This confounds methods like RNA-Seq or metagenomic abundance estimation likewise.

In the following we will address the problem at hand as the *repeat resolution problem*, no matter where the closely related reads stem from.

12.2 Problem definition

In a previous lecture we saw a correction method for genomic abundances that worked well even for closely related genomes, but is not able to assign an individual read to its correct location (in a repeat copy resp. a different genome).

In this lecture we formalize the problem of *repeat resolution* and show a statistical method to find signals that point to differences that are due to repeats and not due to sequencing errors.

Assume we have collected a set of pairwise overlapping reads and formed a multialignment with them (i.e. a *contig*).

The presented method is based upon the fact that errors in a repetitive contig and the errors in a non-repetitive contig are differently distributed. In a non-repetitive contig errors in overlaps can be explained by sequencing errors which should occur independently from each other in each read.

In contrast to this, repetitive contigs by definition consist of reads that are from instances of a repeat from different genomic locations. Depending on the nature of a repeat, two instances differ from each other by a certain amount.

The following figure shows sequencing errors (in red) and microheterogeneity of a collapsed repeat (in blue).

```
...AGCCGTCAGA...
...AGCCGTCAGA...
...AGCCCTCTGA...
...TGTCGTCTGA...
...AGTCGTCTCA...
...AGTCGTCTGA...
```

The columns in blue are called *separating* columns (by Kececioglu) or *Defined Nucleotide Positions (DNPs)* by Tammi. It is clear that these positions can be used to a) determine, whether there is a compressed repeat, and b) to resolve the compression into the different repeat copies.

There is a number of algorithmic problems one has to address in repeat resolution:

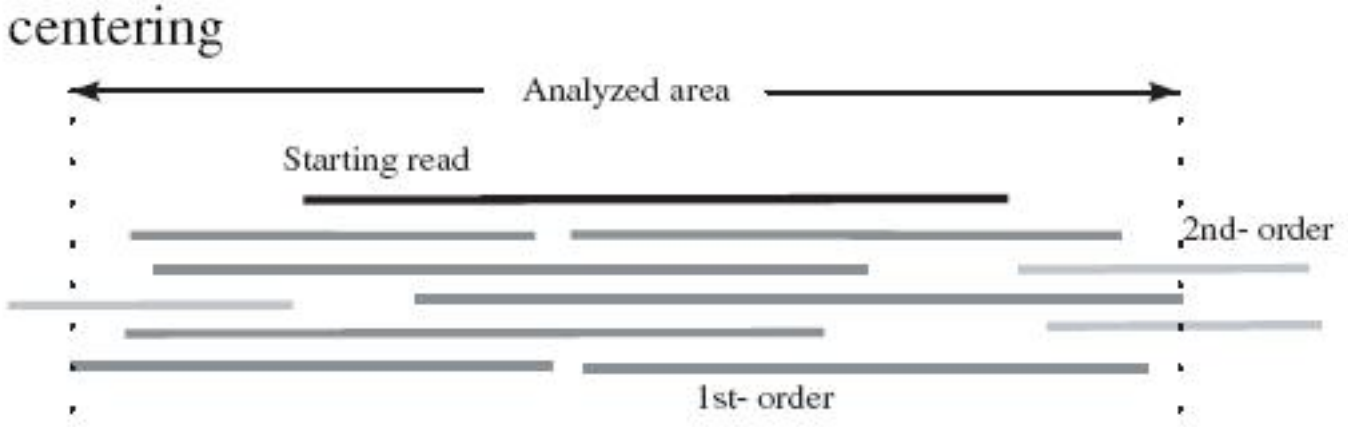
1. *Locating* the positions in the layout where there is a possible compression due to repeats.
2. *Bounding* the region for the analysis of DNPs.
3. *Identifying* the DNPs.
4. *Separating* the set of fragments into subsets belonging to different instances of a repeat.

We will concentrate on the third and fourth point and quickly go over the first two.

Locating the region can be done by using statistical test involving the coverage in a layout (for example by comparing abundances to corrected abundances), or by examining divergent overlaps,

For bounding the region of analysis we introduce a reasonable approach choosen by Tammi. They choose a seed read and form a multi-alignment using all 1st order and 2nd order overlaps. The alignment is optimizied using the program Realigner to remove hopefully most of the alignment errors.

The next picture shows this procedure.



12.3 Finding DNPs

After the multialignment is computed we want to determine the DNPs. To do this both, Kececioglu and Tammi, propose a simple method.

For Kececioglu a *candidate* separating column in an alignment of depth d is one in which a pair of rows agrees on a character with frequency at most $\lfloor d/2 \rfloor$. A candidate column is supported by another column if they are candidates on the same pair of rows. The number t of mutually supporting columns is the *support* of the separating column.

Tammi has a similar approach except that they do not require only a pair of positions for support but D_{min} many positions. However, they need only one column for support.

These approaches work reasonable well, however they neglect one important piece of information available, the quality values for a sequence read. Recall that the quality values encode the probability that the base is indeed correct. Tammi at all address this case, although in a limited fashion.

12.4 Definitions

Consider two fixed positions u and v in an alignment. All definitions for u are similar for v.

Let $a_{u,j}$ be the base at position u in for the j-th sequence. Let $I_{u,j}$ be the indicator for the event that the base at position u of the read j deviates from consensus. If this is the case, the variable is 1 otherwise 0. The total number of deviations from consensus at position u is $N_u = \sum_{j=1}^{k} I_{u,j}$.

Let $I_j = I_{u,j} I_{v,j}$ be the indicator for a *coincidence* in the j-th sequence. Finally, $C = \sum_{j=1}^{k} I_j$ is the total number of coincidences.

The authors then assume independence of the deviation from consensus which is clearly not always true but yields an approximation.

12.5 An exact formula

Tammi derives the distribution of C given the observed values $N_u = n_u$ and $N_v = n_v$. They argue that an exact formula is very complicated if the error probabilities are unevenly distributed. However, if one assumes that all p are uniformly distributed, then one can apply standard combinatorics to derive that C given N_u and N_v is hypergeometrically distributed. Or written differently:

$$P(C = x) = \frac{\binom{n_v}{x}\binom{k-n_v}{n_u-x}}{\binom{k}{n_u}},$$
$$0 \le x \le n_v,\ 0 \le n_u - x \le k - n_v.$$

This is true, since when all ps are identical, each possible configuration has equal probability.

By considering the n_v deviations from the consensus in position v as fixed, the denominator above gives the total number of ways to distribute n_u deviations among k sites, and the numerator the number of ways to do this resulting in x coincidences.

As a reminder: the standard definition of a hypergeometric distribution is: In a bucket are N balls, M of which are white and $N - M$ are black. If you draw n balls without returning them, then the probability that you have k white balls is:

$$P_k(N, M, n) = \frac{\binom{M}{k}\binom{N-M}{n-k}}{\binom{N}{n}},$$

So the conditional distribution is easy to compute if all ps are equal, which they are not. Hence Tammi argues now as follows: Since all values of p are quite small, the unconditional distribution of C is well approximated by a Poisson distribution. In addition, the conditioning on N_u and N_v only introduces weak dependencies, which implies that the Poisson approximation should still be satisfactory.

Hence we have to compute the mean parameter of the Poisson distribution

$$\lambda = E(C \mid N_u = n_u, N_v = n_v).$$

12.6 The Poisson approximation

From the definition of C follows:

$$E(C \mid N_u = n_u, N_v = n_v) = \sum_{j=1}^{k} E(I_{u,j} = 1 \mid N_u = n_u) E(I_{v,j} = 1 \mid N_v = n_v).$$

Furthermore it holds:

$$P(I_{u,j} = 1 \mid N_u = n_u) = \frac{P(I_{u,j} = 1, N_u = n_u)}{P(N_u = n_u)}$$

$$= \frac{P(I_{u,j} = 1, N_u^{(j)} = n_u - 1)}{P(I_{u,j} = 1, N_u^{(j)} = n_u - 1) + P(I_{u,j} = 0, N_u^{(j)} = n_u)}$$

where $N^{(j)} = N_u - I_{u,j}$ denotes the total number of deviations from consensus at site u excluding read j.

Note that $I_{u,j}$ and $N_u^{(j)}$ are independent. Furthermore N_u and $N_u^{(j)}$ are approximately Poisson distributed. Let $\lambda_u = \sum_{i=1} p_{u,i}$ and $\lambda_u^{(j)} = \lambda_u - p_{u,j}$, respectively, denote the means of those distributions. It follows that

$$\frac{P(I_{u,j} = 1, N_u^{(j)} = n_u - 1)}{P(N_u = n_u)}$$

is approximately

$$\left(p_{u,j} e^{-\lambda_u^{(j)}} {\lambda_u^{(j)}}^{n_u - 1} / (n_u - 1)!\right) /$$

$$\left(p_{u,j} e^{-\lambda_u^{(j)}} {\lambda_u^{(j)}}^{n_u - 1} / (n_u - 1)! + (1 - p_{u,j}) e^{-\lambda_u^{(j)}} {\lambda_u^{(j)}}^{n_u} / n_u!\right)$$

This becomes approximately

$$\frac{P(I_{u,j} = 1, N_u^{(j)} = n_u - 1)}{P(N_u = n_u)} \approx \frac{n_u p_{u,j}}{n_u p_{u,j} + \lambda_u^{(j)}(1 - p_{u,j})}.$$

A corresponding result applies to $P(I_{v,j} = 1 \mid N_v = n_v)$ and we conclude that $E(C \mid N_u = n_u, N_V = n_v)$ is approximately

$$\sum_{j=1}^{k} \left(\frac{n_u p_{u,j}}{n_u p_{u,j} + \lambda_u^{(j)}(1 - p_{u,j})} \times \frac{n_v p_{v,j}}{n_v p_{v,j} + \lambda_v^{(j)}(1 - p_{v,j})} \right).$$

Hence, the suggested approximation of the distribution of C given $N_u = n_u$ and $N_v = n_v$, is to approximate it with the Poisson distribution having the mean specified above.

This again implies that the hypothesis that coincidences occur by chance rather than for systematic reasons can be tested by comparing the observed values of c_{obs} with what to expect from the derived approximate distribution.

We compute $p^{corr} = 1 - \sum_{i=0}^{c_{obs}-1} po(i)$, where $po(i)$ is the probability function for the above Possion variable with mean $E(C \mid N_u = n_u, N_V = n_v)$. p_{corr} is the probability of observing c_{obs} or more coincidences between columns u and v. The hypothesis is accepted if p^{corr} is greater than p_{max}^{corr}.

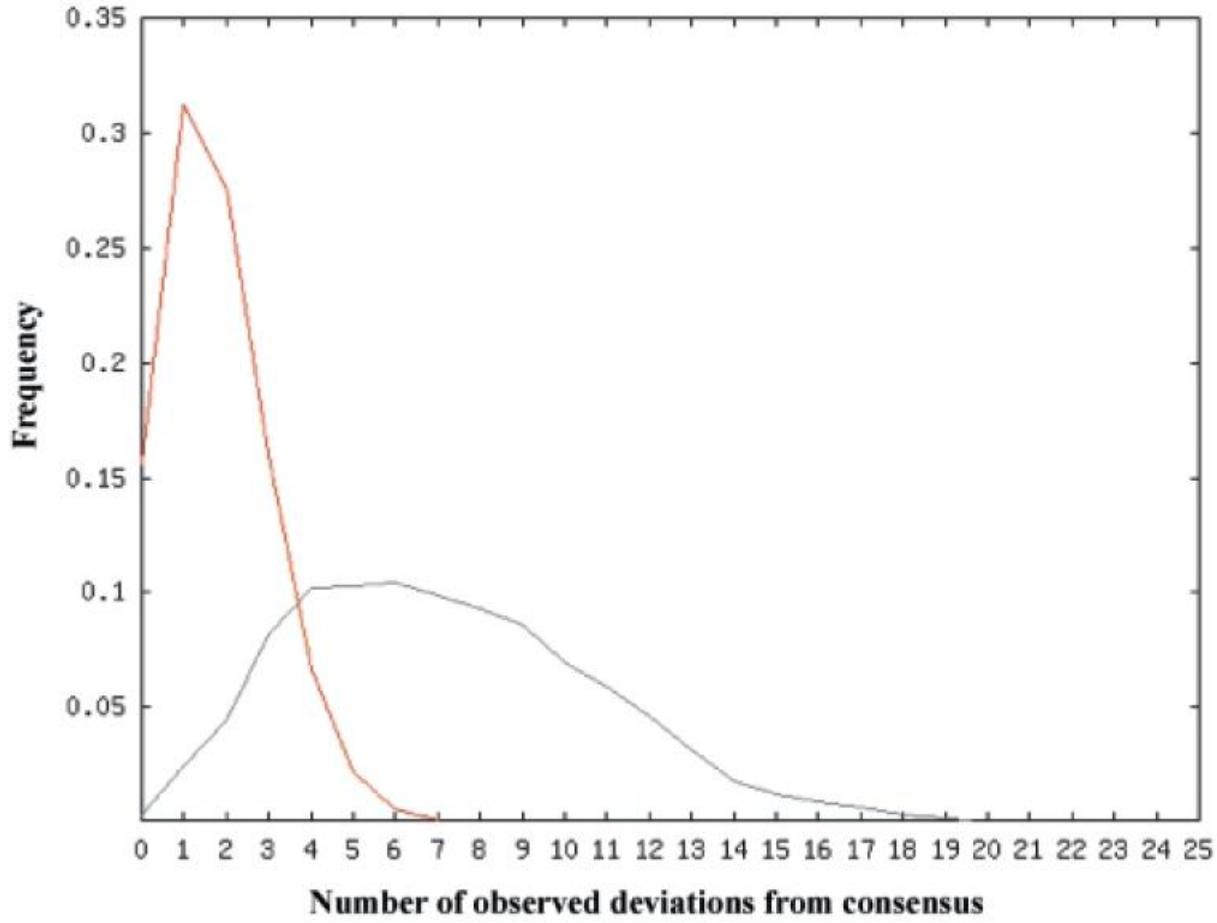
(If there are columns with a large number of differences Tammi gives a possible correction.)

12.7 Results of Tammis methods

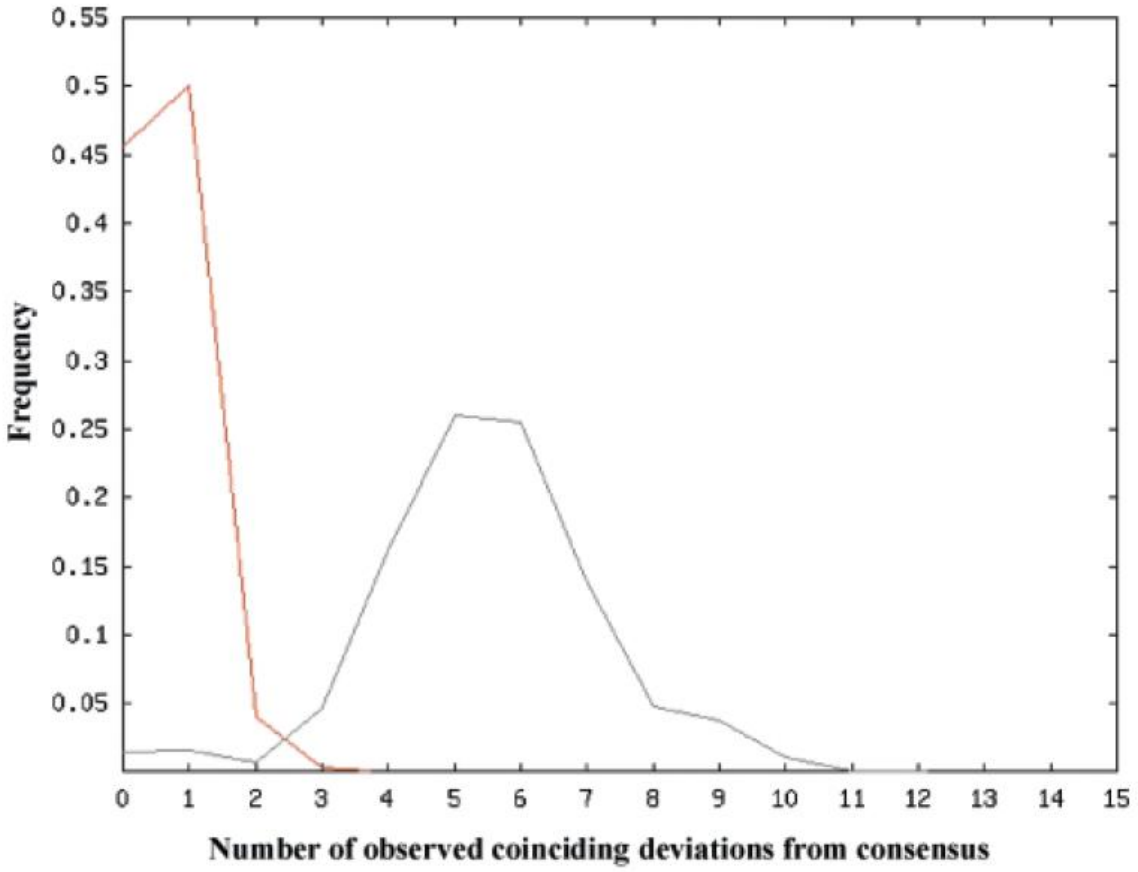
In order to evaluate the method, five sets of simulations were performed. Real quality value files from shotgun data was used and for those repeats were artifically introduced, namely of length 1000, 2000, 3000 bases repeated 4, 6, 8 and 10 times in tandem. The repeats differ 1 percent from each other

The first three sets of simulations differ only in the amount of sequencing error (via quality trimming). The average error rates are 4.3, 3.3 and 2.6 percent, respectively. Set 4 and 5 each have an average quality of 2.6 and differ in the coverage (3.5 to 10.2).

First we look at the separation power of the number of observed deviations from consensus and plot those caused by sequencing errors together with those caused by real differences. (Data sim3 with 10 tandem repeats).



Now we look at the separation power of the number of observed coincidences and plot those caused by sequencing errors together with those caused by real differences.



As expected, it is not sufficient to look only at single columns in the sequence alignment, whereas for coincidences the distributions are clearly separated for $D_{min} > 2$.

Next we look at the error rate (i.e. how often do I call a DNPs when it was not a DNP) and the sensitivity (How many of the real DNPs did I find) of the different situations (ignore the value S_T).

Here the error rate and sensitivity of Tammi's basic method.

Table 2. The results of the basic method in Sims 1–3. The error, ε, sensitivity in respect to true differences in reads, S_R, and sensitivity in respect to differences in the template, S_T, at different $D_{\min}$. S_R is set to 100 in Sim 1, $D_{\min} = 2$, for comparison

	$D_{\min}$				
Simulation	2 (ε) S_R/S_T (%)	3 (ε) S_R/S_T (%)	4 (ε) S_R/S_T (%)	5 (ε) S_R/S_T (%)	6 (ε) S_R/S_T (%)
Sim 1	(59) 100/97	(4.3) 89/87	(0.55) 81/78	(0.42) 71/65	(0.36) 60/53
Sim 2	(40) 81/94	(2.6) 73/82	(0.37) 64/70	(0.28) 55/57	(0.26) 43/43
Sim 3	(25) 71/90	(0.71) 62/76	(0.27) 54/62	(0.21) 44/48	(0.20) 33/34

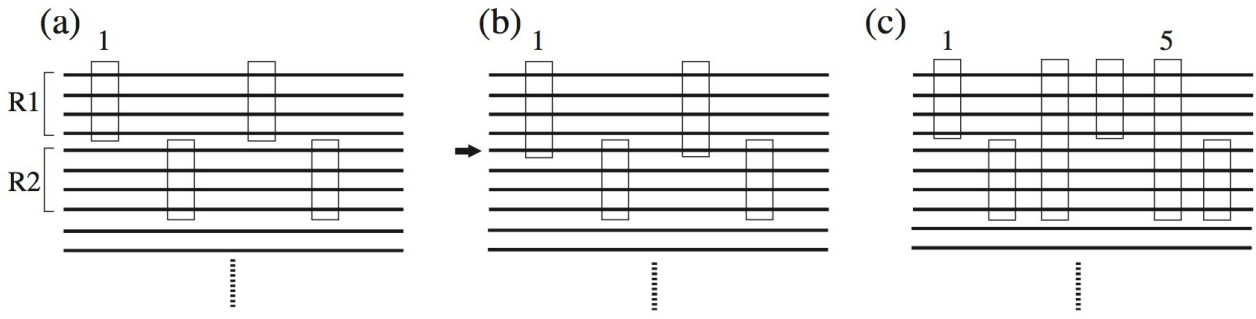
We look now at the error rate and sensitivity of Tammi's extended method first.

Table 5. The results of the extended method in Sims 1–3. The error at $p_{\max}^{\text{tot}} = 10^{-3}$ ($\varepsilon_{10^{-3}}$), and corresponding sensitivity with respect to true differences in reads (S_R), computed at different $D_{\min}$. S_R is computed in relation to Sim 1, $D_{\min} = 2$ with the basic method

	$D_{\min}$				
	2	3	4	5	6
Simulation	$\varepsilon_{10^{-3}}$ (S_R) (%)	$\varepsilon_{10^{-3}}$ (S_R) (%)	$\varepsilon_{10^{-3}}$ (S_R) (%)	$\varepsilon_{10^{-3}}$ (S_R) (%)	$\varepsilon_{10^{-3}}$ (S_R) (%)
Sim 1	8.9 (91)	1.6 (87)	0.53 (81)	0.42 (71)	0.36 (60)
Sim 2	6.6 (76)	0.92 (72)	0.37 (64)	0.28 (55)	0.21 (43)
Sim 3	5.5 (67)	0.52 (62)	0.27 (54)	0.21 (0.44)	0.20 (0.33)

12.8 Separating repeat copies

Now we have identified a number of DNPs. How do we use them to separate the reads into groups belonging to the different repeat copies?



Here Kececioglu took a rigorous approach. Assume that there are k copies of a repeat. Hence, we ideally would like to partition the reads into k classes $P_1, P_2, \ldots, P_k$ together with k consensus strings $S_1, S_2, \ldots, S_k$ such that the overall number of errors $\sum_{1 \le i \le k} \sum_{F \in P_i} D(F, S_i)$ is minimized.

This function is hard to compute but we can approximate it nicely by a) only considering DNP columns, and b) by choosing one of the strings in the partition as consensus string (this is reasonable, since there are not many DNPs and hence it is likely that one sequence of a group has indeed the consensus characters at the DNP positions).

Thus our objective function is to find a partition into k groups that minimizes:

$$\sum_{1 \le i \le k} \min_{F^* \in P_i} \{ \sum_{F \in P_i} H(F, F^*) \}$$

where $H(\cdot, \cdot)$ is the Hamming distance.

The above problem can be cast as a graph theoretical problem.

If we consider a complete, edge weighted graph K_n (vertices correspond to the reads, edges are weighted with the Hamming distance), our task is to find k star centers and an edge set that spans all vertices such that the overall weight of all choosen edges is minimized.

This problem can be formulated as an ILP and solved by a branch-and-bound algorithm. Given a K_n the ILP has $n^2 + n$ variables:

- for each ordered pair (i, j), where i and j are vertices in K_n, there is a variable x_{ij}.
- for each vertex i in K_n there is a variable y_i.

The ILP wants to minimize $\sum_{i \neq j} w_{ij} x_{ij}$ and has a $O(n^2)$ constraints:

- $\forall i$ and j, $x_{ij} \ge 0$,
- $\forall i$, $y_i \ge 0$,
- $\forall j$, $\sum_{1 \le i \le n} x_{ij} \ge 1$
- $\forall i$ and j, $y_i \ge x_{ij}$, and
- $\sum_{1 \le i \le n} y_i = k$

This ILP is solved by computing the LP relaxation and a simple application of the branch-and-bound paradigm.

Integer solutions in each node of the enumeration tree are computed by rounding in such a way, that those k vertices are choosen as star centers that have the heighest average fractional weight by adjacent edges.

As branching variables those x_{ij} are choosen whose fractional value is nearest to 0.5. (If the x_{ij} are integral so are the y_j.)

- Repeat resolution is an important practical problem arising in shotgun sequence assembly projects.
- It can be divided into four subproblems: Locating the repeat region, Bounding the area of analysis, Identifying DNPs, and Separating repeat copies.
- Tammi et al propose a statistical method that perfoms well for identifying DNPs
- Kececioglu gives a nice formulation to solve the separation problem.
- So far no approach uses mate pairs as additional information.