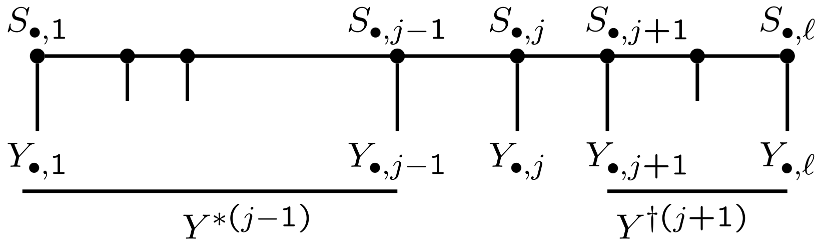
4.4.1 The backwards Baum algorithm and $\Pr(S_{\bullet,j} \mid \mathbf{Y})$:



- Now also define

$$R_j^{\dagger}(s) \;=\; \Pr(Y_k, k=j,\ldots,\ell \mid S_{\bullet,j}=s) = \Pr(Y^{\dagger(j)} \mid S_{\bullet,j}=s).$$

- $$\begin{aligned} R_j^{\dagger}(s) &= \Pr(Y_k, k=j,\ldots,\ell \mid S_{\bullet,j}=s) \\ &= \sum_{s^*} \Pr(Y_k, k=j,\ldots,\ell, S_{\bullet,j+1}=s^* \mid S_{\bullet,j}=s) \\ &= \Pr(Y_j \mid S_{\bullet,j}=s) \sum_{s^*} R_{j+1}^{\dagger}(s^*) \Pr(S_{\bullet,j+1}=s^* | S_{\bullet,j}=s) \end{aligned}$$

- Then $\Pr(S_{\bullet,j}=s \mid \mathbf{Y}) \;=\; \dfrac{\Pr(\mathbf{Y}, S_{\bullet,j}=s)}{\Pr(\mathbf{Y})} \;=\; \dfrac{R_j^*(s) R_j^{\dagger}(s)}{\Pr(\mathbf{Y})}$

4.4.2 Expected recombination counts: implementing EM:

- Recall from 4.2.2 we want

$$\tilde{X}_j \;=\; \mathbf{E}(X_j \mid \mathbf{Y}) \;=\; \sum_i \mathbf{E}(|S_{i,j+1} - S_{i,j}| \mid \mathbf{Y})$$

- Note that

$$\begin{aligned} \Pr(S_{\bullet,j}, S_{\bullet,j+1} \mid \mathbf{Y}) &= \Pr(S_{\bullet,j}, S_{\bullet,j+1}, \mathbf{Y}) / \Pr(\mathbf{Y}) \text{ and} \\ \Pr(S_{\bullet,j}, S_{\bullet,j+1}, \mathbf{Y}) &= \Pr(Y^{*(j-1)}, S_{\bullet,j}) \Pr(Y_{\bullet,j} \mid S_{\bullet,j}) \\ &\quad \Pr(S_{\bullet,j+1} \mid S_{\bullet,j}) P(Y^{\dagger(j+1)} \mid S_{\bullet,j+1}) \end{aligned}$$

- The first term is just the $R_j^*(S_{\bullet,j})$ we had in the forwards Baum algorithm, the second is just a single-locus probability of data given inheritance, the third is just the recombination/non-recombination transitions in I_j interval, and the final is $R_{j+1}^{\dagger}(S_{\bullet,j+1})$ from the backwards version of the Baum algorithm.

- On small pedigrees, the EM map estimation can be implemented.

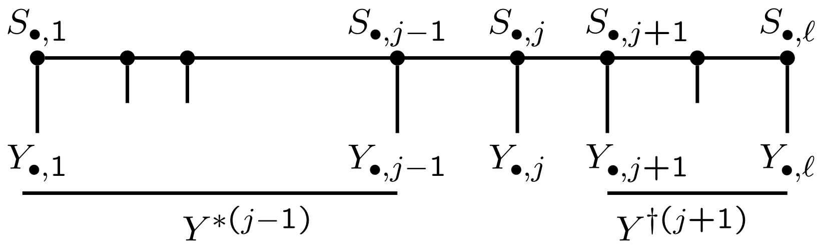
4.4.3 The joint pattern of S over loci:

• 4.4.1 gives us probabilities of $S_{\bullet,j}$ given $\mathbf{Y}$ and hence probabilities of *ibd* at each locus j. Each $S_{\bullet,j}$ can take 2^m values.

• 4.4.2 gives us pairwise probabilities of $(S_{\bullet,j}, S_{\bullet,j+1})$, and hence expected recombination counts, given $\mathbf{Y}$. Each $(S_{\bullet,j}, S_{\bullet,j+1})$ can take $2^m \times 2^m = 4^m$ values.

• But suppose we want $\mathbf{S}$ jointly over all the loci; this is infeasible to compute eactly, even on small pedigrees. $\mathbf{S}$ can take $2^{m\ell}$ values – there are too many possible $\mathbf{S}$.

4.4.4 Monte Carlo realization of S given Y:



• Compute $R_j^*(s) = \Pr(Y^{*(j)}, S_{\bullet,j} = s)$, $j = 1, 2, 3, ...\ell$ as before.

• First, $S_{\bullet,\ell}$ is sampled from $\propto \Pr(\mathbf{Y}, S_{\bullet,\ell}) = \Pr(Y_{\bullet,\ell}|S_{\bullet,l})R_\ell^*(S_{\bullet,\ell})$.

(All sampling probabilities will be normalized over 2^m s-values.)

• Then, given a realization of $(S_{\bullet,j+1} = s^*, S_{\bullet,j+2}, \ldots, S_{\bullet,\ell})$,

$$\Pr(S_{\bullet,j} = s \mid S_{\bullet,j+1} = s^*, S_{\bullet,j+2}, \ldots, S_{\bullet,\ell}, \mathbf{Y}) \;=$$
$$\Pr(S_{\bullet,j} = s|S_{\bullet,j+1} = s^*, Y^{*(j)}) \;\propto\; \Pr(S_{\bullet,j+1} = s^*|S_{\bullet,j} = s)$$
$$R_j^*(s)\Pr(Y_{\bullet,j}|S_{\bullet,j})$$

• Normalizing these probabilities, we realize each $S_{\bullet,j-1}$, for $j = \ell, \ell - 1, \ldots, 4, 3, 2$ in turn, providing an overall realization $\mathbf{S} = (S_{\bullet,1}, \ldots, S_{\bullet,\ell})$ from $\Pr(\mathbf{S} \mid \mathbf{Y})$.

4.4.5 Monte Carlo estimates of *ibd* and genetic maps:

• Instead of computing $\Pr(S_{\bullet,j} \mid \mathbf{Y})$ we can sample it, and hence get estimates of *ibd* patterns at each locus j.

• We can make estimates of *ibd* jointly over loci – for example, the probability that an individual is autoygous over a set of loci, not just the separate probability of each.

• Intead of computing

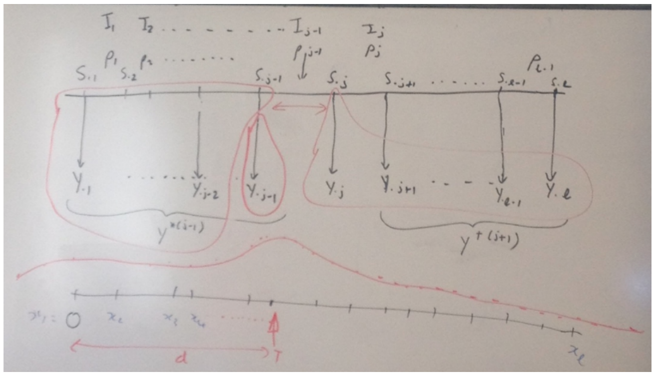
$$\tilde{X}_j \;=\; \mathbf{E}(X_j \mid \mathbf{Y}) \;=\; \sum_i \mathbf{E}(|S_{i,j+1} - S_{i,j}| \mid \mathbf{Y})$$

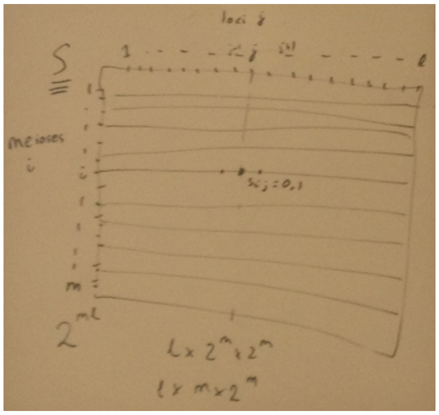
we can count recombination events in N realized $\mathbf{S}$ for all map intervals and each gender.

• Hence we can do Monte Carlo EM, replacing the E-step by these Monte Carlo estimates at each stage.

• Generally, Monte Carlo EM works as well as regular EM, at least for the initial steps. Initially, the Monte Carlo sample size N need not be large, although for the final EM steps it should be increased.

Photos from Qian:


I1 I2 . . . . I j-1 Ij
ρ1 ρ2 . . . . ρ j-1 ρj ρℓ-1
S.1 S.2 S.j-1 S.j S.j+1 . . . . . . S.ℓ-1 S.ℓ
Y.1 . . . . . . Y.j-2 Y.j-1 Yj Y.j+1 . . . Yℓ-1 Y.ℓ
Y*(j-1)
Y†(j+1)
x1 = 0 x2 x3 x4
d
T
xℓ

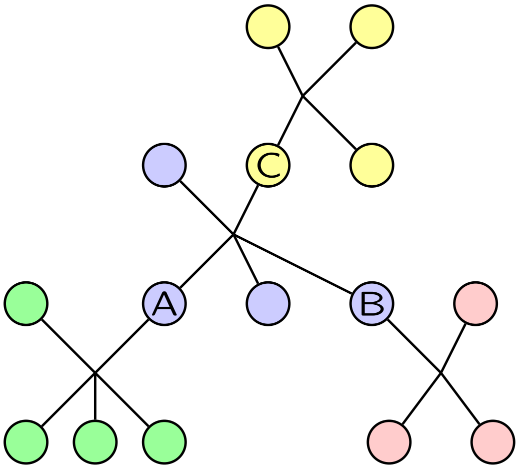


• Left: Above: Showing forwards (R^*) and backwards ($R^\dagger$) components of the HMM likelihood computation. Below: a map-specific lod score computed at locations along a chromosome.

• Right: The grid of $2^{m\ell}$ components $S_{,j}$ of $\mathbf{S}$, showing the Markov dependence across loci j, and the computational complexity of HMM computations.

4.5.1 Conditional independence over pedigree members:

- We get our DNA from our parents and copy to our offspring.
  All meioses (transmissions of DNA) are independent.
  Given the genotypes of parents, sibs are independent, and independent of grandparents, uncles, cousins, ...

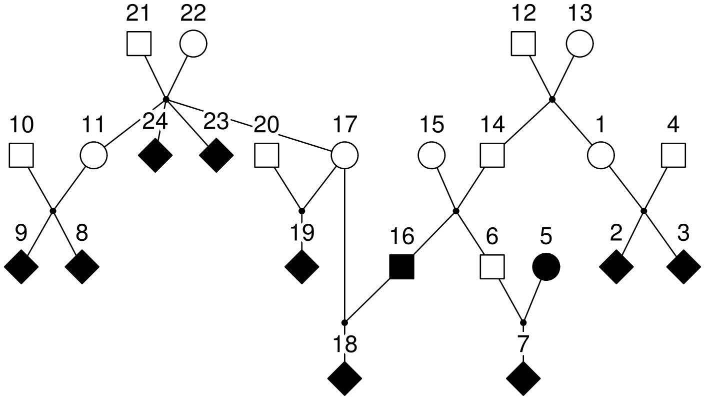


- Sum over the green family for each genotype of A
- Sum over the pink family for each genotype of B
- Using these values, sum over the blue family for each genotype of C.
- Using these values for each C, sum over the yellow family.

- First used by Haldane and Smith (1947) (without computers)
  Formalized (and programmed) by Elston and Stewart (1971).

4.5.2 Elston-Stewart algorithm:

- An example pedigree without loops. Shaded individuals are those for whom phenotypic data are assumed to be available



- Unlike a chromosome, pedigrees have a natural direction; probabilities of a set of offspring given the two parental genotypes is easier than of the parents given the offspring.

4.5.3 Pedigree peeling:

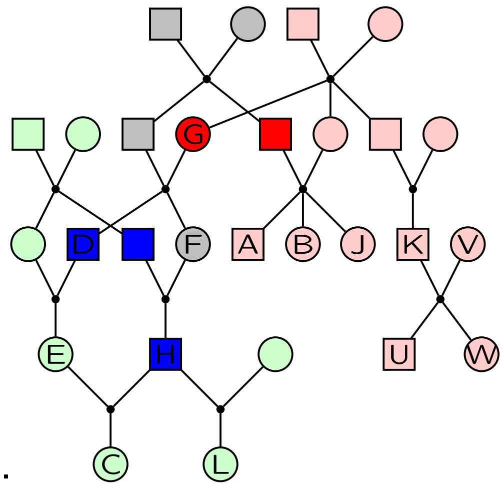
- Similar ideas to HMM underlie pedigree peeling. We don't have Markov chain, but, for pedigrees with no loops, we do have that conditionally of genotype (pair of haplotypes) of each individual, the data "above" are independent of data "below".
- For any individual i Define:

$$\begin{aligned} \text{Forwards/Down: } R_i^*(g) &= \Pr(\text{ data above, } G_i = g) \\ \text{Backwards/Up: } R_i^\dagger(g) &= \Pr(\text{ data below} \mid G_i = g) \end{aligned}$$

- Equations very similar to HMM, at each stage putting in genotype probabilities (if a founder), transmission probabilities from parents to offspring, and probabilities of data marker and/or trait) given the genotype.
- Hence, we can sum over all the unobserved genotypes to compute the probability of trait and/or marker data; that is, the likelihood $\Pr(\mathbf{Y})$.
- Implemented in LIPED, LINKAGE, FASTLINK, VITESSE

4.5.4 Joint pedigree peeling:

- The main problem with pedigree peeling is that the genotypes we sum over, and the transmissions we consider, are of pairs of haplotypes, jointly over loci.
- So computation is now linear in pedigree size but the number of genotypes each individual can have is exponential in number of loci.



- Methods can be extended to pedigrees with loops, but this is even more computationally intensive.
- Given genotypes on a cutset, the data on individuals each side of the cutset are independent. – e.g. the red/pink and blue/green sets.
- Above/below no longer well-defined. Individuals may be Below some but above others in the same cutset.

4.5.5 Monte Carlo given trait data – SIMLINK:

- In early 1980s, often we had trait data; trait data had simple known models (e.g. dominant, recessive)

- Marker maps were just starting; marker typing was expensive. How to persuade NIH to fund a linkage study? How to show the potential data would have power to detect linkage, if present?

- We can simulate trait and marker data jointly quite easily, assuming some trait model. But it is much better to simulate what marker data would look like conditional on the trait data we have.

- Ploughman and Boehnke (1989) solved this problem: see next slide. Their SIMLINK program produced Elods given the observed data, and assuming a marker of some informativeness (e.g. 4 equifrequent alleles), at some distance from the trait locus (e.g. 10 cM).

- If this Elod is "big enough" then this indicates that the trait data are sufficient for marker typing to be worth doing.
For some time, NIH required SIMLINK evidence in grant proposals.

4.5.6 SIMLINK – cf markerdrop for steps (1) to (4):

- (1) Peel up at the trait locus, saving the $R_i^\dagger(g)$ for each trait-locus genotype g of each individual i: $R_i^\dagger(g) = \Pr($ data below $\mid G_i = g)$.

- (2) Assign founder genotypes at the top, in accordance with trait model and upward peeling.

- (3) Simulate genotypes back down at the trait locus, using the saved partial sums — see 4.4.4 for HMM analogy.

- (4) Simulate marker genotypes at loci linked to the trait, at some assumed marker allele frequencies and recombination fraction.

- (5) Do the lod score for each simulated data set.

- (6) Average over simulated data sets to compute an empirical Elod.

Note the analogy to the HMM realization of $\mathbf{S}$ (see 4.4.4). Here we first ccmpute up (backwards in time), and then simlate down (forwards). (Note 2: Could peel any direction, but data are at bottom.)

Note 3: MORGAN/markerdrop is similar, but uses $S_{\bullet,j}$ not genotypes.

4.6.1 What about big pedigrees with many markers??:

• The Lander-Green (or more generally HMM) approach is restricted to relatively small pedigrees ($m \leq 27$).

• The Elston-Stewart approach (or more generally pedigree-peeling) is restricted to just a few loci ($\ell \leq 4$).

• What about m and ℓ both large?

(1) Break up the pedigrees (to reduce m), and pretend they are unrelated. So compute lod scores independently for each piece, and sum them. This loses information.

(2) Break up the chromosome (to very small ℓ) – e.g. interval mapping, in which we use just two markers at a time, and hypothesize trait locus positions between them. This loses information and requires very tedious piecing together many lod-score segments (which don't match at the marker locations).

4.6.2 Monte Carlo Estimation of lod scores:

• This formulation due to Lange and Sobel (1991).
It is similar to version implemented in MORGAN/lm_linkage.

• Let Z be trait data, and $\mathbf{Y}$ the marker data. Let γ be hypothesized position of trait locus (this γ was d in 4.3.4). Then

$$\frac{L(\gamma)}{L(\infty)} = \frac{P_\gamma(Z, \mathbf{Y})}{\Pr(Z).\Pr(\mathbf{Y})} = \frac{P_\gamma(Z \mid \mathbf{Y})}{\Pr(Z)}$$

• Now let $\mathbf{S}$ be the inheritance patterns $S_{\bullet,j}$ at the marker loci.

$$P_\gamma(Z \mid \mathbf{Y}) = \sum_{\mathbf{S}} P_\gamma(Z \mid \mathbf{S}) \Pr(\mathbf{S} \mid \mathbf{Y})$$

• Suppose we could sample $\mathbf{S}$ from $\Pr(\mathbf{S} \mid \mathbf{Y})$: giving realizations $\mathbf{S}^{(k)}$, $k = 1, ..., N$. Then we could estimate $P_\gamma(Z \mid \mathbf{Y})$ by averaging the N values of $P_\gamma(Z \mid \mathbf{S}^{(k)})$. (These computed by pedigree peeling.)

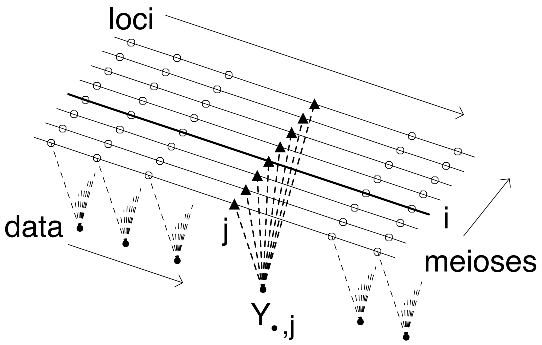
• We can do this with one set of $\mathbf{S}^{(k)}$ for all the hypothesized γ and hence get a Monte Carlo estimate of the map-specific lod score.

4.6.3 MCMC sampling of $\mathbf{S}$ given marker data $\mathbf{Y}$:

- The “SIMLINK” peeling approach (4.5.6) allows us to sample $S_{\bullet,j}$ given $Y_{\bullet,j}$ at a single locus j, but not all loci at once.

- The HMM approach (4.4.4) allows us to sample $\mathbf{S}$ given $\mathbf{Y}$ for a small number of meioses, but not all meioses at once.

- The independence of meioses i and Markov dependence of vectors $S_{\bullet,j}$ provide good MCMC block Gibbs samplers: for ℓ loci and m meioses.



L-sampler: resample $S_{\bullet,j}$ given $\mathbf{Y}$ and $S_{\bullet,j'}, j \neq j'$ Heath 1997.
Uses pedigree computations, $O(\ell m)$.

MM-sampler: resample $\{S_{i,\bullet}; i \in I^*\}$ $|I^*| = m^*$ given $\mathbf{Y}$ and $\{S_{i',\bullet}; i' \notin I^*\}$
Uses HMM computation, $O(\ell m^* 2^{m^*})$, $m^* << m$.
Tong & Thompson 2008

4.6.4 Added notes re MCMC:

- L-sampler is irreducible, but does not mix well.
  Depending on data $\mathbf{Y}$, MM-sampler may not be irreducible.

- Together (random choice at each step) they do fine, and can be used to provide estimates of the map-specific lod score across the chromosome.

- This is the form of MCMC implemented in MORGAN. SIMWALK-2 uses quite a different framework for the MCMC – but the same lod score estimator.

- For older, more spaced, miscrosatellite markers, we compute the lod score at points in marker intervals (see Lab 5). With newer denser SNP markers, we compute at a subset of the marker positions.

- The “lm” of lm_auto and lm_linkage programs in MORGAN indicates these are MCMC programs using the LM-sampler - this combination of locus-sampler (L) and meiosis-sampler (M).

4.6.5 Estimating *ibd* probabilities by MCMC:

• Recall the inheritance patterns $\mathbf{S}$ determine the *ibd* pattern among any set of "proband gametes" (maternal/paternal gametes of specified individuals).

• Since we have MCMC realizations of $\mathbf{S}$ given marker data $\mathbf{Y}$, we can use these to estimate patterns of *ibd* given $\mathbf{Y}$.

• We can estimate at each locus j using $S_{\bullet,j}$, or we can estimate jointly at sets of loci – e,g, over windows of a few loci.

• Data Z at a trait locus t also provide inheritance information (especially i for a rare almost recessive trait, such as in Lab 4).

• It is possible to compute $\Pr(Z \mid S_{\bullet,t})$ —not in this class!
So, we can include the trait locus and trait data Z in the MCMC, although computations are heavier.

• This is what is implemented in MORGAN/lm_auto, either for markers only, or for markers with a simple trait locus included (e.g. Lab 4).

blank slide: