MPRI – Cours 2-12-2

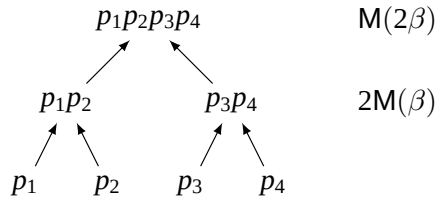
F. Morain

Lecture IV-9: Applications of fast multiplication

2012/10/22

Product trees: principles

Imagine all p_i's have the same size β.



Product tree: $2\mathsf{M}(\beta) + \mathsf{M}(2\beta)$.

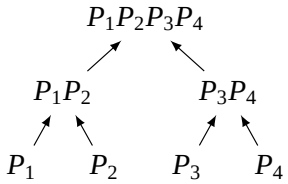
Naive case: $\underbrace{p_1p_2}_{\mathsf{M}(\beta)} + \underbrace{(p_1p_2)p_3}_{\mathsf{M}(2\beta,\beta)} + \underbrace{(p_1p_2p_3)p_4}_{\mathsf{M}(3\beta,\beta)} \approx 6\mathsf{M}(\beta)$.

Comparison: $4\mathsf{M}(\beta)$ vs. $\mathsf{M}(2\beta)$? Equal if $\mathsf{M}(\beta) = \beta^2$, product tree better if $\mathsf{M}(\beta) = \beta^a$, $a < 2$.

General principle: only the last step counts.

With polynomials

Goal: compute $Z = P_1 \cdots P_m$ for polynomials $P_i(X)$.

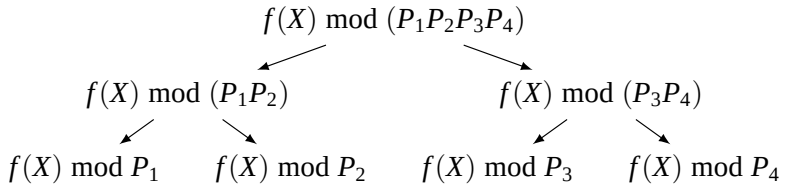


Typical application: PolyFromRoots, i.e., given $(x_i)_{0 \le i < n}$, build $P(X) = \prod(X - x_i)$. Cost is $O(\mathsf{M}(n)\log n)$.

Fast multipoint evaluation

Goal: compute $f(X) \bmod P_i(X) = X - x_i$ for all i.

Use a remainder tree, i.e.,



Key property: $f(X) \bmod (X - x_i) = f(x_i)$

Complexity: $O(\mathsf{M}(n)\log n)$.

Fast resultant and discriminant

$$P(X) = \prod_{i=0}^{n-1}(X - \alpha_i), Q(X) = \prod_{j=0}^{m-1}(X - \beta_j)$$

$$\mathrm{Res}(P, Q) = \prod_{i,j}(\alpha_i - \beta_j) = \prod_{i=0}^{n-1} Q(\alpha_i).$$

Algorithm: use fast multipoint evaluation to compute all $Q(\alpha_i)$; finish with a product tree.

$$\begin{aligned}\mathrm{Disc}(P) &= (-1)^{n(n-1)/2}\mathrm{Res}(P, P') \\ &= \prod_{i<j}(\alpha_i - \alpha_j)^2 \\ &= (-1)^{n(n-1)/2}\prod_{i=0}^{n-1} P'(\alpha_i).\end{aligned}$$

Rem. The resultant can be computed using Euclid's algorithm when the roots are not known.