توری از درخت‌ها (Mesh of Trees)

۲d-MOT

- $N \times N$ پردازنده‌ی برگ
- N درخت سطری (درخت سطر i ام یا $(i,*)$-tree) و N درخت ستونی (درخت ستون j ام یا $(*,j)$-tree)
- ۲N درخت هر کدام با $N - ۱$ پردازنده‌ی غیر برگ (در مجموع ۲$N(N - ۱)$ پردازنده)
- در کل ۳$N^۲$ − ۲N پردازنده
- درجه‌ی برگ‌ها و ریشه‌ها ۲ و درجه گره‌های داخلی ۳

- قطر $4 \lg N$
- پهنای دوبخشی برابر N
- گاهی ریشه‌ی درخت سطر i و ستون i را در هم ادغام می‌کنند.
- ساختار بازگشتی. با حذف ریشه‌ها، ۴ MOT هر کدام با $N/2 \times N/2$ پردازنده ایجاد می‌شود.
- می‌توان با یک $N \times N$ MOT، گراف $K_{N\times N}$ را با درجه‌ی کندی $2 \log N$ شبیه‌سازی کرد.
- به‌دلیل پهنای دوبخشی $O(N)$، مرتب‌سازی N^2 عدد در برگ در کم‌تر از $O(N)$ امکان‌پذیر نیست.
- مسیریابی هم مثل مرتب‌سازی است. اما در حالت خاص می‌توان مسیریابی را در $2 \log N$ مرحله انجام داد.

مسیریابی در $2 \lg N$ مرحله

- اگر $N < M$ بسته در ریشه‌ی درخت‌های سطری بخواهند به ریشه‌های درخت‌های ستونی بروند.

مرتب‌سازی N عدد در $4 \lg N$ مرحله‌ی کلمه‌ای

دو مرحله: محاسبه‌ی مرتبه‌ی عناصر و سپس مسیردهی.

- فرض می‌کنیم که ریشه‌های درخت‌های سطر iام و ستون iام در هم ادغام شده‌اند.
- ریشه‌ی iام x_i را دارد.
- x_i به برگ‌ها ارسال می‌شود.
- در انتها پردازنده‌ی (i, j) x_i و x_j را دریافت می‌کند.
- این پردازنده اگر $x_j \leq x_i$ باشد صفر و گرنه ۱ را ذخیره می‌کند.
- اعداد ۱ در درخت‌های ستونی j با هم جمع می‌شوند و مرتبه‌ی x_j، یا r_i به‌دست می‌آید.
- x_i به ریشه‌ی r_iام مسیردهی می‌شود.

ضرب ماتریس $A[N \times N]$ در بردار x یا $Ax = y$

$$y_i = \sum_{j=۱}^{N} a_{ij} x_j$$

- a_{ij} در برگ (i, j) قرار دارد.
- x_j در ریشه‌ی درخت ستونی j.
- x_j به برگ‌های درخت $(*, j)$ ارسال می‌شود.
- $\Sigma_{j=۱}^{N} a_{ij} x_j$ در ریشه‌ی سطر i محاسبه می‌شود.

۳d-MOT

- $N \times N \times N$ برگ با شماره‌های (i, j, k)
- $N^۲$ عدد درخت بعد ۱ (با درخت‌های $(*, j, k)$)
- $N^۲$ عدد درخت بعد ۲ (با درخت‌های $(i, *, k)$)
- $N^۲$ عدد درخت بعد ۳ (با درخت‌های $(i, j, *)$)
- در مجموع $N^۳ + ۳N^۲(N - ۱)$ پردازنده و $۳N^۲(۲N - ۱)$ یال
- قطر $۶ \lg N$
- پهنای دوبخشی $O(N^۲)$

ضرب دو ماتریس $C = AB$ در ۳d-MOT

- ورودی: a_{ij} در ریشه‌ی درخت $(i, j, *)$
- ورودی: b_{jk} در ریشه‌ی درخت $(*, j, k)$
- a_{ij} برای برگ‌های درخت $(i, j, *)$ ارسال می‌شود.
- b_{jk} برای برگ‌های درخت $(*, j, k)$ ارسال می‌شود.
- پس (i, j, k) هم a_{ij} را دریافت می‌کند و هم b_{jk}.
- برروی درخت $(i, *, k)$ محاسبه‌ی زیر انجام می‌شود:

$$c_{ik} = \sum_{j=1}^{N} a_{ij} b_{jk}$$

- خروجی c_{ik} در زمان ۲ $\lg N$ در ریشه‌ی درخت $(i, *, k)$ به‌دست می‌آید.

درخت فراگیر کمینه در ۲d-MOT

- ورودی: یک گراف وزن‌دار با N رأس
- ساختار: ۲d-MOT که ریشه‌های درخت‌های سطر i و ستون i در هم ادغام شده باشند (r_i).
- برگ (i, j) وزن آن یال یا $w(i, j)$
- r_i شماره‌ی leader(i) یا $L(i)$ را دارد. i leader است اگر $L(i)$ $= i$.

کلیات الگوریتم

الگوریتم بر اساس کروسکال است:

- هر رأس در یک مجموعه قرار دارد که MST آن نهایی و مشخص شده است. هر مجموعه را یک «ابررأس» می‌گوییم. هر ابررأس یک رأس $leader$ دارد. هر رأس شماره‌ی $leader$ خود را دارد.
- $\lg N$ بار تکرار کن

۱) به‌صورت موازی همه‌ی سبک‌ترین یال‌های بین ابررأس‌ها را پیدا کن.

۲) این زوج–ابررأس‌ها را در هم ادغام کن و اطلاعات $leader$ را هم به‌روز کن.

مرحله‌ی ۱: به‌دست آوردن نزدیک‌ترین ابررأس $A(i)$ به هر رأس i و فاصله‌ی آن.

$$w(i, A(i)) = \min_{1 \leq j \leq N} \{w(i,j) | L(i) \neq L(j)\}$$

۱) r_i، i و $L(i)$ را به همه‌ی برگ‌هایش می‌فرستد.

۲) برگ (i,j)، i و $L(i)$، j و $\mathrm{L}(j)$ و نیز $w(i,j)$ را خواهد داشت.

۳) بر روی درخت سطر i اگر $L(i) \neq L(j)$، $w(i,j)$ به بالا ارسال و عمل $\min$ گرفته می‌شود.

۴) در انتها در r_i، $A(i)$، $L(A(i))$ و $w(i, A(i))$ به‌دست می‌آید.

این کار در $2 \lg N$ مرحله انجام می‌شود.

مرحله‌ی ۲: محاسبه‌ی $P(j)$ برای هر $j = L(j)$

$P(j)$ شماره‌ی leader نزدیک‌ترین ابررأس به ابررأس j است. یعنی $P(j)$ مقدار $L(A(i))$ است که مقدار زیر کمینه شود:

$$W(j) = \min\{w(i, A(i)) | L(i) = L(j) = j\}$$

۱) $L(A(i))$ و $w(i, A(i))$ را در درخت سطر i پایین می‌فرستیم.

۲) برگ (i, j)، $L(A(i))$ و $w(i, A(i))$ و نیز $L(A(j))$ و $w(j, A(j))$ را دریافت می‌کند.

۳) در درخت ستونی مربوط به leader محاسبه‌ی فوق انجام می‌شود.

مرحله‌ی ۳: ادغام ابرراُس‌ها

- گرافی بر روی ابرراُس‌ها با یال‌های $(j, P(j))$ را در نظر بگیرید.
- این گراف جهت‌دار و بدون دور است (مگر برای دقیقاً یک دور به طول ۲
- اگر عمل $P(j) \leftarrow P(P(j))$ را $\lg N$ بار انجام دهیم، گراف به‌صورت ستاره در می‌آید.
- مرکز ستاره همان leader جدید است.
- با عمل $L(i) \leftarrow P(L(j))$ که توسط همه‌ی راُس‌ها انجام می‌شود، عمل ادغام کامل می‌شود.

نحوه‌ی انجام $Z(j) \leftarrow X(Y(j))$

۱) $X(i)$ به برگ‌های درخت سطر i ارسال می‌شود.

۲) در همان زمان، درخت ستون j j امین برگ را $Y(j)$ انتخاب می‌کند.

۳) برگ انتخابی $X(Y(j))$ را به ریشه‌ی درخت ستون j ارسال می‌شود.

این کار در $۲ \lg N$ مرحله

مرحله‌ی ۳ در $۲ \lg^۲ N$ و کل الگوریتم در $۲ \lg^۳ N + O(\lg^۲ N)$ انجام می‌شود.