الگوریتم L^3 و کاربردهای آن

غلامرضا برادران خسروشاهی
شاهین آجودانی نمینی، محمد رجبی طرخورانی*

۰. مقدمه

اگر با دانلدکنوت همرأی باشیم که "علم کامپیوتر عمدتاً همان مطالعهٔ الگوریتمهاست" [۹]، باید بپذیریم که با رشد و گسترش روزافزون کامپیوتر، نظریهٔ الگوریتمها یا "ریاضیات الگوریتمی" نیز رشد متزایدی خواهد داشت و این رشد بالمآل در ریاضیات اثراتی ژرف و پایا خواهد گذاشت. در اینجا بر آن نیستیم که بر جنبه‌های مختلف این دیدگاه یا اعتقاد تأکید ورزیم، لکن می‌خواهیم از مصداق بارزی سخن بگوییم که نشانه‌ای مثبت و عمیق از اثرات کامپیوتر بر ریاضیات است.

بحث ما دربارهٔ الگوریتمی با زمان چندجمله‌ای است که اولین بار برای تجزیهٔ یك چندجمله‌ای $f \in \mathbf{Q}[X]$ یك متغیره با ضرایب گویا به عوامل تحویل‌ناپذیر در $\mathbf{Q}[X]$ به‌کار گرفته شد و سپس در خدمت ردکنندگان حدس مرتنس و حل کنندگان "مسائل مجموع زیرمجموعه‌ای"[1]، درآمد و آنگاه درحل دستگاههای دیوفانتی و یافتن جواب ویژهٔ دستگاههای بسیاربزرگ معادلات خطی همگن مؤثر واقع شد.

این الگوریتم توسط لنسترا[2]، لنسترا جونیور[3]، و لواش[4] [۵] تدوین شده و به همین مناسبت به الگوریتم L^3 شهرت یافته‌است.

در این مقدمه، در چارچوب بررسی کلی و تاریخی تجزیهٔ چندجمله‌ایها، سرشت و جایگاه این الگوریتم را کمی روشن می کنیم. در بخش دوم مقاله، الگوریتم را با دقت وجزئیات، همراه با چند مثال ساده تشریح می‌نماییم و در بخشهای بعد به توصیف برخی از کاربردهای آن می‌پردازیم.

به‌مسألهٔ امکان تجزیهٔ چندجمله‌ایها به‌عوامل تحویل‌ناپذیر روی $\mathbf{Q}$ قرنهاست که پاسخ مثبت داده شده است، لکن حل "مؤثر" آن تنها در سالهای اخیر به‌سامان رسیده‌است. نیوتن گویا اولین کسی بوده است که راهی برای یافتن مقسوم‌علیه‌های خطی و درجهٔ دوم پیشنهاد کرده‌است؛ سپس در سال ۱۷۹۳، فریدریش فون شوبرت ستاره‌شناس روش نیوتن‌را تعمیم می‌دهدو تمام عوامل تحویل‌ناپذیر یك چندجمله‌ای را به دست می آورد. روش فون‌شوبرت با نشان دادن تصمیم‌پذیری‌مسألهٔ تجزیهٔ چندجمله‌ایها به‌عوامل تحویل‌ناپذیر، منطقیون را خشنود می‌سازد، لکن به‌علت "کندی"، دیگرانی راکه به‌دنبال حل عملی مسأله هستند راضی نمی کند. این روش برای یك چندجمله‌ای درجهٔ n، دست کم به 2^n مرحله‌نیاز دارد تا نشان دهد که چند جمله‌ای تحویل‌ناپذیر است یا خیر. بنابراین، برای تجزیهٔ چندجمله‌ایهای با درجهٔ بزرگتر از ۲۰ عملی نیست. اساس مطلب در اینجا مسألهٔ پیچیدگی محاسبه است. یك الگوریتم تجزیه تا چه‌اندازه مجاز است که وقتگیر باشد؟ متخصصین کامپیوتر معتقدند که تنها راه‌حلهای با زمان چندجمله‌ای– یعنی الگوریتمهایی که تعداد گامهای اجرای هر یك از آنها نسبت به‌اندازهٔ ورودی یك چندجمله‌ای است– قابل قبول وعملی هستند. روش فون شوبرت برحسب درجهٔ چندجمله‌ای، نمایی است.

چون تجزیهٔ چندجمله‌ایها روی هیأتهای متناهی ساده‌تر است، لذا ابتدا مسألهٔ تجزیه روی هیأتهای‌متناهی مانند $\mathbf{Z}_p$، مورد بحث قرار گرفته و در سال ۱۹۶۷ برلی کمپ[1] الگوریتمی با زمان

1. subset sum problems
2. A. K. Lenstra
3. H. W. Lenstra Jr.
4. Lovasz

1. Berlekamp

چندجمله‌ای برای تجزیهٔ یک چندجمله‌ای از درجهٔ n روی $\mathbf{Z}_p$ ساخته است [۸]. به‌دنبال آن هنزل[1] نحوهٔ انتقال یک تجزیه از $\mathbf{Z}_p$ به $\mathbf{Z}_{p^2}$ را بیان کرد [۱].

حال فرض کنید عدد اول p، مبین (تعریف مبین را در یادداشت پایان بخش ۳ ببینید.) چندجمله‌ای f را عاد نکند و h یک عامل تحویل‌ناپذیر f در $\mathbf{Z}_p[X]$ باشد. ما به‌دنبال عامل تحویل‌ناپذیر h_0 از f در $\mathbf{Z}[X]$ هستیم که بر h تقسیم‌پذیر باشد. شرط تقسیم‌پذیری h_0 بر h به این معناست که h به "شبکهٔ خاصی" تعلق داشته باشد و شرط اینکه f بر h_0 تقسیم‌پذیر باشد این است که ضرایب h_0 نسبتاً کوچک باشند. بنابراین، مسأله به‌این منجر می‌شود که به‌دنبال عنصر "کوچکی" در شبکهٔ به‌دست آمده از h باشیم. درست این کار است کــه توسط الگوریتم L^3 انجام می‌پذیرد. این الگوریتم به دفعات لازم تکرار می‌شود تا کلیهٔ عوامل f در $\mathbf{Z}[X]$ به دست آید. لازم به یادآوری است که تجزیهٔ یک چندجمله‌ای در $\mathbf{Q}[X]$ با تجزیهٔ چندجمله‌ای اولیه‌ای از $\mathbf{Z}[X]$ (یعنی یک چند جمله‌ای که بزرگترین مقسوم‌علیه مشترک ضرایبش برابر ۱ باشد)، معادل است.

به‌طور مجمل، الگوریتم L^3 برای شبکه‌های تعمیم یافتهٔ ریاضیات کلاسیک پایــه‌ای به دست می‌دهد که به وسیلهٔ آن می‌توان عنصر "کوچکی" را در شبکه به دست آورد و این عنصر کوچک است که کارایی بسیار دارد. بدین ترتیب الگوریتمی کــه ظاهر بسیار ساده‌ای دارد، به‌مسائل مهمی پاسخ می‌گوید.

بگذارید در پایان این مقدمه، حرف اول مقالهٔ لانداو[2] [۴] را که تأکیدی است بر اهمیت ریاضیات الگوریتمی متذکر شویم: "علوم کامپیوتر راهی بــرای برگشت به مبدأ ریاضیات، یعنی حساب و محاسبه، فراهم می‌سازد. بــا مطرح شدن مسألهٔ یافتن اعداد اول، تجزیهٔ اعداد بار دیگر به صحنه می‌آید. و حالا نیز داستانی دیگر: تجزیهٔ چندجمله‌ایها به‌عوامل تحویل‌ناپذیر روی اعداد گویا."

۲. الگوریتم پایهٔ تحویل‌یافته برای شبکه

در این بخش ابتدا به ذکر چند تعریف می‌پردازیم. فرض کنید n عددی صحیح و مثبت باشد. زیــر مجموعهٔ L از فضای n بعدی اقلیدسی $\mathbf{R}^n$، یک شبکه نــامیده می‌شود اگر و تنها اگر یک پایهٔ $B=\{b_1,\dots,b_n\}$ از $\mathbf{R}^n$ وجود داشته‌باشد به‌طوری که هرعضو L یک ترکیب خطی صحیح از بردارهای B باشد. به‌عبارت دیگر

$$L=\sum_{i=1}^{n}\mathbf{Z}b_i=\left\{\sum_{i=1}^{n}r_ib_i \mid r_i\in\mathbf{Z},\quad 1\leqslant i\leqslant n\right\}.$$

B یک پایهٔ L و n رتبهٔ L نامیده می‌شود.

یادآوری می‌کنیم که به‌ازای هر پایهٔ $B=\{b_1,\dots,b_n\}$ از $\mathbf{R}^n$، یک پــایهٔ متعامد $B^*=\{b_1^*,\dots,b_n^*\}$ را می‌توان به طور استقرایی از فــرایند متعامدسازی گرام ـ اشمیت به صورت زیر به‌دست آورد:

$$b_i^*=b_i-\sum_{j=1}^{i-1}\mu_{ij}b_j^*,\quad 1<i\leqslant n$$

$$\mu_{ij}=(b_i,b_j^*)/(b_j^*,b_j^*),\quad 1\leqslant j<i\leqslant n$$

کــه در آن $(\cdot\,,\cdot)$ نمایشگر ضرب داخلی معمولی در $\mathbf{R}^n$ است. همچنین دترمینان L کــه با $d(L)$ نشان داده می‌شود به صورت $d(L)=|\det(b_1,\dots,b_n)|$ تعریف می‌شود. b_iها بردارهای ستونی هستند و $d(L)$ به پایهٔ انتخاب شده بستگی ندارد. پــایهٔ مرتب $B=[b_1,\dots,b_n]$ بــرای شبکهٔ L، تحول یافته (یا y - تحویــل یــافته) نــامیده مــی‌شود اگــر شرایــط زیر برقرار باشند:

$$|\mu_{ij}|\leqslant\frac{1}{2},\quad 1\leqslant j<i\leqslant n \qquad (i)$$

$$|b_i^*+\mu_{ii-1}b_{i-1}^*|^2\geqslant y\cdot|b_{i-1}^*|^2\quad 1<i\leqslant n. \qquad (ii)$$

که در اینجا y یــک مقدار ثابت $1/4<y<1$ است و $|\cdot|$ نمایشگر طول اقلیدسی است. لنسترا و همکارانش [۵] الگوریتمی ساختند که یک پایهٔ مرتب $B=[b_1,\dots,b_n]$ شبکهٔ L را به یک پایهٔ تحویل یافتهٔ $B'=[b'_1,\dots,b'_n]$ تبدیل می‌کند.

اساس الگوریتم L^3 بر به‌کارگیری دو نوع تبدیل خطی زیر، به‌تعداد متناهی دفعه، استوار است:

$T1$: به‌ازای $1<i\leqslant k$ و ثابت‌عمومی $y\in(1/4,1)$، اگر $|b_i^*+\mu_{ii-1}b_{i-1}^*|^2\geqslant y|b_{i-1}^*|^2$ برقــرار نباشد، بردار b_i و b_{i-1} را جا به‌جا می‌کنیم.

$T2$: اگر به‌ازای $k>1$، $|\mu_{kk-1}|>1/2$، آنگاه به‌جای b_k، b_k-rb_{k-1} راقرار می‌دهیم که در آن $r=\mathrm{round}(\mu_{kk-1})$ است. $(\mathrm{round}(\mu)$ نزدیکترین عدد صحیح به μ است.)

علت موفقیت دنبالهٔ تبدیلات $T1$ و $T2$ این است که مقادیر قدیمی μ_{ij} و $|b_i^*|^2$ را بدون به‌کارگیری کل فرایند متعامدسازی می‌توان تازه کرد. هنگامی که نتوان هیچ یک از تبدیلات $T1$ و $T2$ را به‌کار گرفت، الگوریتم به پایان می‌رسد. پایهٔ تحویل‌یافتهٔ B' تقریب صحیحی از پایهٔ B^* حاصل از فرایند گرام ـ اشمیت است و حاوی بردار کوتاهی می‌باشد. (قسمتهای (۳) و (۴) قضیهٔ ۱ را ببینید.)

از تعریفهای فوق قضیهٔ زیر به سادگی نتیجه می‌شود که بخشی از خواص پایهٔ تحویل‌یافته را برملا می‌سازد.

قضیهٔ ۱. فــرض کنید L شبکه‌ای در $\mathbf{R}^n$ بــا پایهٔ تحویل یــافتهٔ $B=[b_1,\dots,b_n]$ و پایهٔ متعامد متناظر $B^*=[b_1^*,\dots,b_n^*]$ باشد. در این‌صورت به‌ازای $y=3/4$ احکام زیر برقرارند:

$$|b_j|^2\leqslant 2^{i-1}|b_i^*|^2,\quad 1\leqslant j\leqslant i\leqslant n \qquad (1)$$

$$d(L)\leqslant\prod_{i=1}^{n}|b_i|\leqslant 2^{n(n-1)/4}d(L) \qquad (2)$$

---

1. Hensel 2. S. Landau

$$|b_1| \leqslant 2^{(n-1)/4}\, d(L)^{1/n} \tag{۳}$$

$$|b_1|^2 \leqslant 2^{n-1}|x|^2, \quad x\in L, \quad x\neq 0 \tag{۴}$$

$$|b_j|^2 \leqslant 2^{n-1}\cdot \max\{|x_1|^2, \dots, |x_t|^2\} \tag{۵}$$

که در اینجا $j=1, 2, \dots, t$ و $x_j\in L$ (x_jها مستقل خطی‌اند).

اثبات این احکام وسایر گزاره‌های زیر در سطح مقدماتی است. به [۵] مراجعه کنید. در جدول ۱ تمامی الگوریتم و در تابلوی ۱ فلوچارتی از آن ارائه شده است.

جدول ۱. الگوریتم L^3

$$\left.\begin{array}{l} b_i^* := b_i; \\ \left.\begin{array}{l} \mu_{ij} := (b_i, b_j^*)/B_j; \\ b_i^* := b_i^* - \mu_{ij} b_j^* \end{array}\right\} j = 1, 2, \dots, i-1; \\ B_i := (b_i^*, b_i^*) \end{array}\right\} i = 1, 2, \dots, n;$$

$k := 2;$

(۱) perform (*) for $l = k-1$;

if $B_k < \left(\frac{3}{4} - \mu_{kk-1}^2\right) B_{k-1}$, go to (۲);

perform (*) for $l = k-2, k-3, \dots, 1$;

if $k = n$, terminate;

$k := k+1$;

go to (۱);

(۲) $\mu := \mu_{kk-1}$; $B := B_k + \mu^2 B_{k-1}$; $\mu_{kk-1} := \mu B_{k-1}/B$;

$B_k := B_{k-1} B_k / B$; $B_{k-1} := B$;

$$\begin{pmatrix} b_{k-1} \\ b_k \end{pmatrix} := \begin{pmatrix} b_k \\ b_{k-1} \end{pmatrix};$$

$$\begin{pmatrix} \mu_{k-1j} \\ \mu_{kj} \end{pmatrix} := \begin{pmatrix} \mu_{kj} \\ \mu_{k-1j} \end{pmatrix} \text{ for } j = 1, 2, \dots, k-2;$$

$$\begin{pmatrix} \mu_{ik-1} \\ \mu_{ik} \end{pmatrix} := \begin{pmatrix} 1 & \mu_{kk-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & -\mu \end{pmatrix} \begin{pmatrix} \mu_{ik-1} \\ \mu_{ik} \end{pmatrix} \text{ for } i = k+1, \dots, n;$$

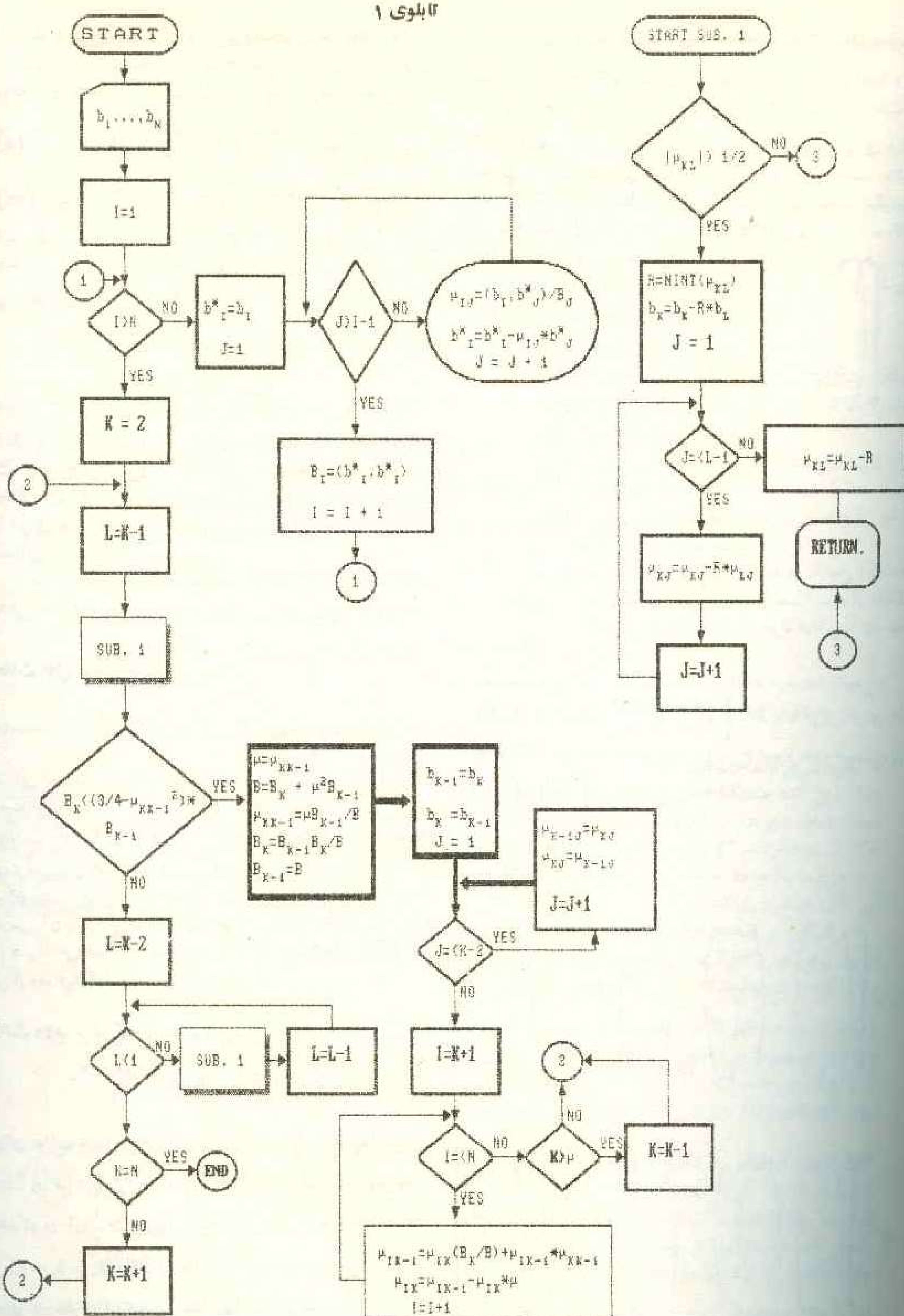
if $k > 2$, then $k := k-1$;

go to (۱).

(*) if $|\mu_{kl}| > 1/2$, then;

$$\begin{cases} r := \text{integer nearest to } \mu_{kl};\ b_k := b_k - r b_l; \\ \mu_{kj} := \mu_{kj} - r\mu_{lj} \text{ for } j = 1, 2, \dots, l-1; \\ \mu_{kl} := \mu_{kl} - r \end{cases}$$

تابلوی ۱


START
b1,...,bN
I=1
I>N
NO
YES
b*I=bI
J=1
J>I-1
μIJ=(bI,b*J)/BJ
b*I=b*I-μIJ*b*J
J = J + 1
BI=(b*I,b*I)
I = I + 1
K = 2
L=K-1
SUB. 1
BK<(3/4-μKK-1^2)*BK-1
μ=μKK-1
B=BK + μ^2BK-1
μKK-1=μBK-1/B
BK=BK-1BK/B
BK-1=B
bK-1=bK
bK =bK-1
J = 1
μK-1J=μKJ
μKJ=μK-1J
J=J+1
J=<K-2
L=K-2
L<1
L=L-1
I=K+1
I=<N
K>μ
K=K-1
K=N
END
K=K+1
μIK-1=μIK(BK/B)+μIK-1*μKK-1
μIK=μIK-1-μIK*μ
I=I+1
START SUB. 1
|μKL|> 1/2
R=NINT(μKL)
bK=bK-R*bL
J = 1
J=<L-1
μKL=μKL-R
μKJ=μKJ-R*μLJ
RETURN.
J=J+1

حال به شرح الگوریتم L^3 می‌پردازیم. در هر گام الگوریتم با یک پارامتر k سروکار داریم که $k\in\{1,2,\dots,n+1\}$. در ابتدا $k=2$. در هر گام شرایط زیر باید برقرار باشند:

$$|\mu_{ij}|\leqslant \frac{1}{2},\quad 1\leqslant j<i<k \qquad (*)$$

$$|b_i^*+\mu_{ii-1}b_{i-1}^*|^2\geqslant y\cdot|b_{i-1}^*|^2,\quad 1<i<k. \qquad (**)$$

این شرایط به وضوح به ازای $k=2$ برقرارند و وقتی $k=n+1$، پایهٔ تحویل‌یافته به دست می‌آید. پس فرض می‌کنیم $k\leqslant n$.

ابتدا شرط (*) را بررسی می‌کنیم. اگر به ازای $k>1$،

$$|\mu_{kk-1}|>\frac{1}{2}$$

آنگاه فرض می‌کنیم $r=\text{round}\,(\mu_{kk-1})$ و b_k-rb_{k-1} را به جای b_k قرار می‌دهیم. به‌ازای $j<k-1$، $\mu_{kj}-r\mu_{k-1j}$ را به جای μ_{kj}، و $\mu_{kk-1}-r$ را به جای μ_{kk-1} قرار می‌دهیم. با این‌کار شرط (*) برقرار می‌شود. اثبات این مدعا نیز درسطح مقدماتی است.

حال شرط (**) را درنظر می‌گیریم و دربارهٔ دوحالت زیر به‌طور جداگانه بحث می‌کنیم.

حالت اول. فرض کنید $k\geqslant 2$ و

$$|b_k^*+\mu_{kk-1}b_{k-1}^*|^2<\frac{3}{4}|b_{k-1}^*|^2.$$

در این صورت b_k را با b_{k-1} تعویض می‌کنیم و به بقیهٔ b_iها دست‌نمی‌زنیم. حال بردارهای b_{k-1}^* و b_k^* و اعداد μ_{kk-1}، μ_{k-1j}، μ_{kj}، μ_{ik-1}، μ_{ik} به‌ازای $j<k-1$ و $i>k$ تغییر می‌یابند. مهمترین این تغییرات، قرار گرفتن $b_k^*+\mu_{kk-1}b_{k-1}^*$ به جای b_{k-1}^* است. بدین ترتیب مقدار جدید $|b_{k-1}^*|^2$ از $3/4$ مقدار قدیمی آن بیشتر می‌شود. برای انجام تغییرات، به‌جای k، $k-1$ را قرار می‌دهیم. از اینجا دوباره به شرایط (*) و (**) باز‌می‌گردیم و الگوریتم را پی می‌گیریم.

حالت دوم. فرض کنید $k=1$ یا

$$|b_k^*+\mu_{kk-1}b_{k-1}^*|^2\geqslant\frac{3}{4}|b_{k-1}^*|^2.$$

دراین حالت باید به‌ازای $1\leqslant j\leqslant k-1$، $|\mu_{kj}|\leqslant\frac{1}{2}$ برقرار باشد. در غیر این صورت، فرض کنید l نزدیکترین اندیس به k باشد که در آن $|\mu_{kl}|>\frac{1}{2}$. حال قرارمی‌دهیم $r=\text{round}(\mu_{kl})$ و $b_k=b_k-rb_l$، وکار را مانند حالت اول ادامه می‌دهیم تا به ازای $1\leqslant j\leqslant k-1$، $|\mu_{kj}|\leqslant\frac{1}{2}$ برقرار شود.

حال k را با $k+1$ عوض می‌کنیم، و الگوریتم را ادامه می‌دهیم.

قضیهٔ ۲. الگوریتم L^3 به‌پایان می‌رسد هرگاه نتوان تبدیلات $T1$ و $T2$ را به‌کار برد. به‌عبارت دیگر الگوریتم L^3 پایان‌پذیر است.

قضیهٔ ۳. فرض کنید $B=[b_1,\dots,b_n]$ پایهٔ مرتبی برای شبکهٔ صحیح L باشد به‌طوری که به ازای $1\leqslant i\leqslant n$ داشته باشیم $|b_i|\leqslant M$، که M یک عدد ثابت است. در آن صورت الگوریتم L^3 پایهٔ تحویل‌یافتهٔ $B'=[b_1',\dots,b_n']$ برای L را با حداکثر $O(n^4\log_2 M)$ عمل حسابی تولید می‌کند، و اعداد صحیحی که این عملیات در آنها انجام می‌گیرد حداکثر دارای طول $O(n\log_2 M)$ هستند.

به‌طور خلاصه:
الگوریتم L^3 هنگامی‌که روی یک پایهٔ Bی یک شبکهٔ n بعدی $L\subset\mathbb{Z}^n$ اعمال‌شود آن را به‌یک پایهٔ تحویل‌یافتهٔ B' برای L تبدیل می‌کند. ضمناً

(۱) برای این کار حداکثر به $O(n^4\log_2 M)$ عمل نیاز است.

(۲) B' تقریباً متعامد است (تقریبی صحیح از پایهٔ متعامدسازی گرام‌ـاشمیت است).

(۳) B' بردارهای کوتاهی در بر دارد. درعمل ثابت‌شده است که طول بردارهای بدست آمده از اعمال الگوریتم بسیار کوتاهتر از طول ادعایی آنهاست.

مثال ۱. فرض کنید $B=\left[\begin{pmatrix}5\\0\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix}\right]$ پایهٔ مرتبی برای یک شبکهٔ $L\subset\mathbb{Z}^2$ باشد. این پایه به‌وضوح تحویل‌یافته نیست، ولی $B'=\left[\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}5\\0\end{pmatrix}\right]$ تحویل‌یافته است.

مثال ۲.

$$B=\left[\begin{pmatrix}1\\0\\1\\1\\1\end{pmatrix},\begin{pmatrix}1\\2\\0\\1\\1\end{pmatrix},\begin{pmatrix}1\\1\\1\\0\\1\end{pmatrix},\begin{pmatrix}1\\1\\2\\2\\0\end{pmatrix},\begin{pmatrix}0\\1\\1\\1\\1\end{pmatrix}\right]$$

$$B'=\left[\begin{pmatrix}0\\1\\0\\-1\\0\end{pmatrix},\begin{pmatrix}-1\\0\\0\\1\\0\end{pmatrix},\begin{pmatrix}1\\1\\-1\\0\\0\end{pmatrix},\begin{pmatrix}1\\1\\1\\0\\1\end{pmatrix},\begin{pmatrix}1\\1\\1\\0\\-1\end{pmatrix}\right]$$

پایهٔ متعامد $(B')^*$ متناظر با B' به‌این صورت است

$$(B')^* = \left[\begin{pmatrix} 1.00 \\ 0.00 \\ 0.00 \\ -1.00 \\ 0.00 \end{pmatrix}, \begin{pmatrix} -1.00 \\ 0.50 \\ 0.00 \\ 0.50 \\ 0.00 \end{pmatrix}, \begin{pmatrix} 0.66 \\ 0.66 \\ -1.00 \\ 0.66 \\ 0.00 \end{pmatrix}, \begin{pmatrix} 0.57 \\ 0.57 \\ 1.14 \\ 0.57 \\ 1.00 \end{pmatrix}, \begin{pmatrix} 0.35 \\ 0.35 \\ 0.70 \\ 0.35 \\ -1.39 \end{pmatrix} \right]$$

$\mu_{21} = -0.50$

$\mu_{31} = 0.50 \qquad \mu_{32} = -0.33$

$\mu_{41} = 0.50 \qquad \mu_{42} = -0.33$

$\mu_{43} = 0.14$

$\mu_{51} = 0.50 \qquad \mu_{52} = -0.33$

$\mu_{53} = 0.14 \qquad \mu_{54} = 0.39$

این محاسبات به‌وسیلهٔ یك برنامهٔ کامپیوتری انجام گرفته است [۱۱]. مرجع اصلی این بخش، [۵] است.

۳. تجزیهٔ سریع چندجمله‌ایها

از آنجایی که خاستگاه اصلی الگوریتم L^3 تجزیهٔ چندجمله‌ایها است، جادارد این مسأله را با تفصیل بیشتری بررسی کنیم تا نحوهٔ به‌کارگیری الگوریتم L^3 در تجزیه دقیقاً روشن‌شود. نخست‌رابطهٔ بین تجزیه وشبکه‌ها را به‌اختصار شرح می‌دهیم. این‌مطالب برای فهم الگوریتم ضروری است.

فرض کنید p یك عدد اول و k یك عدد صحیح مثبت باشد. همچنین‌فرض کنید $f \in \mathbf{Z}[X]$ یك‌چندجمله‌ای ازدرجهٔ n، $n > 0$ و $h \in \mathbf{Z}[X]$ یك چندجمله‌ای با ویژگیهای زیر باشد.

(۱) h یك چندجمله‌ای یکانی است.
(۲) $h(\bmod p^k)$ در $\mathbf{Z}_{p^k}[X]$، $f(\bmod p^k)$ را عاد می‌کند.
(۳) $h(\bmod p)$ در $\mathbf{Z}_p[X]$ تحویل‌ناپذیر است.
(۴) $[h(\bmod p)]^2$ در $\mathbf{Z}_p[X]$، $f(\bmod p)$ را عاد نمی‌کند.

قضیهٔ ۱. چندجمله‌ای $f \in \mathbf{Z}[X]$ دارای عـامل تحویل‌ناپـذیر $h_0 \in \mathbf{Z}[X]$ است اگر $h(\bmod p)$، که در آن h دارای ویژگیهای فوق است، $h_0(\bmod p)$ را عادکند، واین‌عامل صرف‌نظر ازعلامت به‌طور منحصر به فرد تعیین می‌شود. به‌علاوه، اگر g در $\mathbf{Z}[X]$، f را عادکند، آنگاه سه شرط زیر هم‌ارزند:

(الف) $h(\bmod p)$ در $\mathbf{Z}_p[X]$، $g(\bmod p)$ را عاد می‌کند.
(ب) $h(\bmod p^k)$ در $\mathbf{Z}_{p^k}[X]$، $g(\bmod p^k)$ را عاد می‌کند.
(پ) h_0 در $\mathbf{Z}[X]$، g را عاد می‌کند.

تبصرهٔ ۱. فرض کنید m یك عدد صحیح ثابت است به‌طوری که $l \leqslant m$، و نیز فرض‌کنید L مجموعهٔ تمام چندجمله‌ایهای متعلق به $\mathbf{Z}[X]$ با درجهٔ نابیشتر از m است به‌طوری‌که وقتی به‌پیمانهٔ p^k حساب‌شوند بر $h(\bmod p^k)$ در $\mathbf{Z}_{p^k}[X]$ تقسیم‌پذیر باشند. این مجموعه زیر مجموعه‌ای از فضای برداری $(n+1)$ بعدی

$$\mathbf{R} + \mathbf{R} \cdot x + \cdots + \mathbf{R} \cdot x^m$$

است. این فضای برداری با $\mathbf{R}^{m+1}$ تحت نگاشت خطی

$$\sum_{i=0}^{m} a_i x^i \longrightarrow (a_0, \ldots, a_m)$$

یکریخت است. به‌این‌جهت طول‌چندجمله‌ای $f = \sum_{i=0}^{m} a_i x^i$ را به صورت $|f| = \left(\sum_{i=0}^{m} a_i^2\right)^{1/2}$ تعریف می‌کنیم. به‌سادگی می‌توان دیدکه L یك شبکه در $\mathbf{R}^{m+1}$ است وباتوجه به‌شرط (۱) مجموعهٔ

$$\{p^k x^i \mid 0 \leqslant i < l\} \cup \{h x^j \mid 0 \leqslant j \leqslant m - l\}$$

یك پایهٔ L است و ضمناً $d(L) = p^{kl}$.

قضیهٔ ۲. فرض کنید h_0 همان چندجمله‌ای مذکور در قضیهٔ ۱ و L شبکهٔ فوق باشد. نیز فرض‌کنید $b \in L$ در شرط

$$p^{kl} > |f|^m \cdot |b|^n$$

صدق‌کند. دراین صورت b بر h_0 در $\mathbf{Z}[X]$ تقسیم‌پذیر است، وبه‌ویژه $\gcd(f, b) \neq 1$.

قضیهٔ ۳. فرض‌کنید p یك‌عدد اول و k یك‌عدد صحیح مثبت باشد و $f \in \mathbf{Z}(X)$ و $n = \deg f$ و چندجملـه‌ای h در شرایط (۱)، (۲)، (۳) و (۴) صدق کند. همچنین فرض کنید h_0 در شرایط قضیهٔ ۱ صادق باشد، و m و L همان m و L مذکور در تبصرهٔ ۱ باشند. نیز تصور کنید $[b_1, \ldots, b_{m+1}]$ یـك پایهٔ تحویل‌یافته برای L است و

$$p^{kl} > 2^{mn/2} \binom{2m}{m}^{n/2} |f|^{m+n}.$$

دراین‌صورت، $\deg h_0 \leqslant m$ اگر و تنها اگر

$$|b_1| < \left(\frac{p^{kl}}{|f|^m}\right)^{1/n}.$$

قضیهٔ ۴. فرض‌کنید نمادها و فرضها همانند قضیهٔ قبل باشند وعلاوه برآن فرض‌کنیدکـه اندیسی چون $j \in \{1, \ldots, m+1\}$ وجود داشته باشد به‌طوری‌که

$$|b_j| < \left(\frac{p^{kl}}{|f|^m}\right)^{1/n}. \tag{1}$$

حال‌فرض‌کنید t بزرگترین عدد j باخاصیت‌فوق باشد. دراین‌صورت

$$\deg(h_0)=m+1-t$$

$$h_0=\gcd(b_1,\dots,b_t)$$

و نامساوی (۱) برای تمام jها، $1\leqslant j\leqslant t$، برقرار است.

تبصرهٔ ۲. اگر $t=1$، آنگاه b_1 یك عامل تحویل‌ناپذیر f است و نیازی به‌محاسبهٔ بزرگترین مقسوم‌علیه مشترك نیست.

شرح الگوریتم. فرض کنید $f\in\mathbb{Z}[X]$ یك چندجمله‌ای اولیه‌از درجهٔ n ($n>0$) باشد. می‌خواهیم الگوریتمی ارائه کنیم که f را به عوامل تحویل‌ناپذیر در $\mathbb{Z}[X]$ تجزیه کند. الگوریتم شامل دو بخش کمکی و یك بخش اصلی است.

بخش کمکی ۱. فرض کنید‌که علاوه‌بر f و n، یك عدد اول p، یك‌عدد صحیح‌مثبت k و یك‌چندجمله‌ای $h\in\mathbb{Z}[X]$ داده‌شده‌اند و شرایط (۱)، (۲)، (۳)، و (۴) برقرارند. نیز فرض کنید‌که ضرایب h در $\mathbb{Z}_{p^k}$ در شرط

$$|h|^2\leqslant 1+lp^{2k}$$

صدق کنند، که در آن $l=\deg h$. به‌علاوه تصور کنید $m\geqslant l$ داده شده است و نامساوی

$$p^{kl}>2^{mn/2}\binom{2m}{m}^{n/2}|f|^{m+n}$$

برقرار است. در این صورت الگوریتمی‌که ارائه می‌دهیم تصمیم می‌گیرد که آیا $\deg h_0\leqslant m$ یاخیر (h_0 همان‌چندجمله‌ای‌مذکور در قضیهٔ ۱ است). و اگر $\deg h_0\leqslant m$، h_0 را می‌توانیم تعیین کنیم.

فرض‌کنید L شبکهٔ تعریف شده در تبصرهٔ ۱ با پایهٔ

$$\{p^kx^i\mid 0\leqslant i<l\}\cup\{hx^j\mid 0\leqslant j\leqslant m-l\}$$

است. با استفاده از الگوریتم L^3 پایهٔ تحویل یافتهٔ $[b_1,\dots,b_{m+1}]$ را برای L به‌دست می‌آوریم.

اگر $|b_1|>(p^{kl}/|f|^m)^{1/n}$، آنگاه برطبق قضیهٔ ۳ داریم $\deg h_0>m$ و الگوریتم متوقف می‌شود.

اگر $|b_1|<(p^{kl}/|f|^m)^{1/n}$، آنگاه طبق قضیهٔ ۳ و قضیهٔ ۴ داریم

$$\deg h_0\leqslant m,\ h_0=\gcd(b_1,\dots,b_t)\ .$$

در اینجا همان شرایط مذکور در قضیهٔ ۴ روی t برقرارند. (این بزرگترین مقسوم‌علیه مشترك را می‌توان باکاربرد مکرر الگوریتم زیر براینده[1] به دست آورد. (الگوریتم زیر براینده در [۱] آمده است.))

قضیهٔ ۵. تعداد اعمال حسابی لازم در الگوریتم کمکی فوق $O(m^4k\log p)$ است، و اعداد صحیحی که روی آنها این اعمال صورت می‌گیرد هریك دارای طول دوتایی $O(mk\log p)$ است. (منظور از طول دوتایی، تعداد ارقام عدد در دستگاه دوتایی است).

1. subresultant

بخش کمکی ۲. فرض کنید علاوه‌بر f و n، یك‌عدد اول p و یك چندجمله‌ای $h\in\mathbb{Z}[X]$ داده شده‌اند به‌طوری‌که شرایط (۱)، (۲)، (۳)، و (۴) برقرارند با این تفاوت‌که به‌جای k، ۱ قرار دارد. تصور کنید که ضرایب h به $\mathbb{Z}_p$ منتقل شده‌اند. الگوریتمی که اینك عرضه می‌کنیم h_0 را تعیین می‌کند. h_0 یك‌عامل تحویل‌ناپذیر f است به‌طوری که $h(\bmod p)$، $h_0(\bmod p)$ را عاد می‌کند.

فرض کنید $l=\deg(h)$. اگر $l=n$، آنگاه $h_0=f$ و الگوریتم متوقف می‌شود. اگر $n>l$، نخست کوچکترین عدد مثبت k را به‌دست‌می‌آوریم که به‌ازای‌آن، نابرابری (۱) برقرار باشد. با این شرط که به‌جای m عدد $n-1$ قرار گیرد، داریم

$$p^{kl}>2^{(n-1)n/2}\binom{2(n-1)}{n-1}|f|^{2n-1}.$$

سپس h را اصلاح می‌کنیم بدون‌اینکه $h(\bmod p)$ را عوض‌کنیم. این کار را به‌طریقی انجام می‌دهیم که علاوه‌برشرایط (۱)، (۳)، (۴)، شرط (۲) برای مقداری از k که محاسبه می‌شود برقرار باشد. این عمل با استفاده از لم هنزل صورت می‌گیرد. می‌توانیم فرض‌کنیم‌که ضرایب h به $\mathbb{Z}_pk$ منتقل شده‌اند.

گیریم u بزرگترین عدد صحیحی باشد که به ازای آن $l\leqslant(n-1)/2^u$. الگوریتم کمکی ۱ را متوالیاً برای هریك از مقادیر

$$m,\ \left[\frac{n-1}{2^u}\right],\ \left[\frac{(n-1)}{2^{u-1}}\right],\ \dots,\ \left[\frac{(n-1)}{2}\right],\ n-1$$

به کار می‌بریم. ($[x]$ بزرگترین عدد صحیح نابیشتر از x است.) اما به‌محض این‌که برای یکی‌از مقادیر m، الگوریتم کمکی ۱ موفق به‌تعیین h_0 شود، الگوریتم را متوقف می‌سازیم. اگر به‌ازای هیچ‌یك از مقادیر m، h_0 تعیین نشود، آنگاه $\deg h_0>n-1$، و بنابراین $f=h_0$ و الگوریتم متوقف می‌شود.

قضیهٔ ۶. فرض کنید $m_0=\deg h_0$، درجهٔ عامل تحویل‌ناپذیر h_0 از f باشدکه توسط‌الگوریتم‌کمکی ۲ به‌دست‌می‌آید. دراین‌صورت تعداد عملیات حسابی لازم در این الگوریتم مساوی $O(m_0(n^5+n^4\log|f|+n^3\log p))$ است واعداد صحیحی که‌این‌عملیات روی‌آنها انجام می‌گیرد هرکدام دارای طول دوتایی $O(n^3+n^2\log|f|+n\log p)$ است.

بخش اصلی الگوریتم. حال الگوریتمی را شرح می‌دهیم که یك چندجمله‌ای اولیهٔ داده‌شدهٔ $f\in\mathbb{Z}[X]$ با درجهٔ $n>0$ را به عوامل تحویل‌ناپذیر در $\mathbb{Z}[X]$ تجزیه می‌کند.

اولین گام محاسبهٔ براینده، $R(f,f')$، است که در آن f' مشتق f است [برای تعریف براینده به یادداشت پایان بخش ۳ مراجعه کنید.] اگر $R(f,f')=0$، آنگاه f و f' دارای‌بزرگترین مقسوم‌علیه مشترك g در $\mathbb{Z}[X]$ با درجهٔ مثبت است و g را با الگوریتم زیر براینده به دست می‌آوریم. این حالت را در پایان مورد بحث قرار می‌دهیم. پس فرض کنید $R(f,f')\neq 0$.

در گام دوم کوچکترین عدد اول p را به‌دست می‌آوریم که $R(f,f')$ را عاد نکند، و سپس $f(\bmod p)$ را در $\mathbb{Z}_p[X]$

توسط الگوریتم برلی کمپ تجزیه می‌کنیم. توجه دارید که $R(f, f')$ صرف‌نظر از علامت، برابر است با حاصلضرب ضریب پیشرو f درمبین آن. بنابراین $R(f,f')\equiv 0 \pmod p$ نتیجه می‌دهد که $f \pmod p$ هنوز دارای درجهٔ n است و در $\mathbb{Z}_p[X]$ دارای هیچ عامل چندگانه نیست؛ بنابراین شرط (۴) برای هر عامل تحویل‌ناپذیر $h \pmod p$ از $f \pmod p$ در $\mathbb{Z}_p[X]$ برقرار است.

در گام سوم، فرض می‌کنیم که یک تجزیهٔ $f=f_1f_2$ در $\mathbb{Z}[X]$ را می‌شناسیم به قسمی که تجزیهٔ کامل f_1 در $\mathbb{Z}[X]$ و $f_2 \pmod p$ در $\mathbb{Z}_p[X]$ معلوم هستند. در نقطهٔ شروع می‌توانیم فرض کنیم که $f_1=1$ و $f_2=f$. در این وضعیت، کار را به صورت زیر ادامه می‌دهیم: اگر $f_2=\pm 1$، آنگاه $f=\pm f_1$ به‌طور کامل در $\mathbb{Z}[X]$ تجزیه می‌شود، و الگوریتم متوقف می‌شود. حال فرض کنید f_2 دارای درجهٔ مثبت است. در $\mathbb{Z}_p[X]$ یک عامل تحویل‌ناپذیر $h \pmod p$ از $f_2 \pmod p$ را انتخاب می‌کنیم. می‌توانیم فرض کنیم که ضرایب h به‌پیمانهٔ p تحویل یافته‌اند و h دارای ضریب پیشرو ۱ است. اکنون درهمان وضعیتی هستیم که الگوریتم کمکی ۲ را شروع کردیم، با این تفاوت که f_2 نقش f را دارد، و با استفاده از این الگوریتم، عامل تحویل‌ناپذیر h_0 از f_2 را در $\mathbb{Z}[X]$ به دست می‌آوریم؛ زیرا $h \pmod p$، $h_0 \pmod p$ را عاد می‌کند. اکنون f_1 و f_2 را به ترتیب با f_1h_0 و f_2/h_0 عوض می‌کنیم وازفهرست عاملهای تحویل‌ناپذیر $f_2 \pmod p$ آن عاملهایی راکه $h_0 \pmod p$ را عاد می‌کنند حذف می‌کنیم. حال دوباره به آغاز مرحلهٔ سوم برمی‌گردیم. بدین ترتیب توصیف الگوریتم درحالتی که $R(f, f')\neq 0$ است پایان می‌یابد.

اکنون فرض کنید که $R(f, f')=0$، و نیز تصور کنید که g بزرگترین مقسوم‌علیه مشترک f و f' در $\mathbb{Z}[X]$ است، و قرار دهید $f/g=f_0$. در این صورت، f_0 دارای هیچ‌عامل چندگانه در $\mathbb{Z}[X]$ نیست. بنابراین، $R(f_0, f_0')\neq 0$ و می‌توانیم f_0 را توسط قسمت اصلی الگوریتم تجزیه کنیم. چون هرعامل تحویل‌ناپذیر g در $\mathbb{Z}[X]$، f_0 را عاد می‌کند، می‌توانیم تجزیهٔ $f=f_0g$ را با تقسیمهای متوالی کامل کنیم.

قضیهٔ ۷. الگوریتم فوق هر چندجمله‌ای اولیهٔ $f\in\mathbb{Z}[X]$ با درجهٔ مثبت n را به‌عوامل تحویل‌ناپذیر در $\mathbb{Z}[X]$ تجزیه می‌کند. تعداد عملیات حسابی لازم برای اجرای این برنامه $O(n^6+n^5\log|f|)$ است و اعداد صحیحی که این عملیات روی آنها انجام می‌گیرند هرکدام دارای طول دوتایی $O(n^3+n^2\log|f|)$ است.

یادداشت

تعریف ۱. فرض کنید $f\in K[X]$ یک چندجمله‌ای از درجهٔ $n\geqslant 2$ است، و فرض کنید $f(x)=a_0\prod_{i=1}^{n}(x-\alpha_i)$ و α_iها متعلق به‌هیأت شکافندهٔ f روی K هستند. دراین صورت مبین f که با $D(f)$ نمایش داده می‌شود عبارت است از

$$D(f)=a_0^{2n-2}\prod_{1\leqslant i<j\leqslant n}(\alpha_i-\alpha_j)^2.$$

از تعریف $D(f)$ برمی‌آید که f دارای ریشهٔ چندگانه است اگر و تنها اگر $D(f)=0$.

تعریف ۲. اگر $f(x)=a_0x^n+\dots+a_n$ و بپذیریم که ممکن است $a_0=0$، در این صورت n مساوی درجهٔ $f(x)$ نیست. از n به‌عنوان درجهٔ صوری f یاد می‌کنیم.

فرض کنید K یک هیأت و $f\in K[X]$ و

$$g\in K[X] \text{ و } g(x)=b_0x^m+\dots+b_m$$

دو چندجمله‌ای با درجات صوری n و m باشند. دراین صورت برایند این دو چندجمله‌ای که با $R(f, g)$ نمایش داده می‌شود به‌صورت زیر تعریف می‌شود:

$$R(f,g)=\left|\begin{array}{cccc} a_0 & a_1\cdots a_n & & 0\cdots 0\\ 0 & a_0a_1\cdots a_n & & 0\cdots 0\\ & & \vdots & \\ 0\cdots 0 & & a_0 & \cdots a_n\\ b_0 & b_1\cdots b_m & & \\ 0 & b_0\cdots b_1\cdots b_m & & \\ \vdots & & & \\ 0 & 0\cdots 0 & b_0b_1\cdots b_m & \end{array}\right| \begin{array}{l} \left.\begin{array}{l} \\ \\ \\ \\ \end{array}\right\} m\ \text{سطر} \\ \left.\begin{array}{l} \\ \\ \\ \\ \end{array}\right\} n\ \text{سطر} \end{array}$$

این دترمینان از مرتبهٔ $m+n$ است. اگر $a_0\neq 0$ و اگر α_iها ریشه‌های $f(x)$ در هیأت شکافندهٔ f روی K باشند، آنگاه

$$R(f, g)=a_0^m\prod_{i=1}^{n}g(\alpha_i).$$

با توجه به‌اینکه عضو $b\in K$ یک ریشهٔ چندگانهٔ $f\in K[X]$ است اگر وتنها اگر b ریشهٔ مشترکی از f و f' باشد، ارتباط بین $D(f)$ و $R(f, f')$ به‌دست می‌آید.

$$D(f)=(-1)^{n(n-1)/2}a_0^{-1}R(f, f').$$

مراجع عمدهٔ این بخش عبارت‌اند از [۵]، [۴]، [۱]، [۶].

۴. حدس مرتنس و اثبات نادرستی‌آن

رد حدس مرتنس با استفاده ازالگوریتم L^3، در واقع اولین کاربرد مهم این الگوریتم به‌غیر از کاربرد اصلی آن یعنی تجزیهٔ چندجمله‌ایها می‌باشد. این کاربرد برکارایی الگوریتم بیشتر صحه می‌گذارد. حدس مرتنس به‌دلیل اینکه درست بودنش می‌توانست درستی فرضیهٔ ریمان را نتیجه بدهد، در نظریهٔ اعداد از اعتباری برخوردار بود. ولی نادرستی حدس مرتنس چیزی دربارهٔ فرضیهٔ ریمان بیان نمی‌کند.

برای اینکه ارزش این کاربرد بهتر معلوم شود کمی در بارهٔ حدس مرتنس گفتگو می‌کنیم.

* * *

یکی از مسائل مهم در نظریهٔ اعداد، مسألهٔ توزیع اعداد اول است. در این باره "قضیهٔ اعداد اول" می‌گوید که اگر $\pi(x)$ نمایشگر تعداد اعداد اول کوچکتر از x باشد، آنگاه

$$\lim_{x\to\infty}\frac{\pi(x)}{x/\log x}=1.$$

این حدس را اولین بار گاوس و لژاندر بیان کردند و بعدها به‌وسیلهٔ پواسون و آدامار به‌اثبات رسید. ریمان نیز در مقاله‌ای به‌بررسی اعداد اول پرداخت و با فرمول دقیقی توانست $\pi(x)$ را به‌صفرهای تابع زتا مرتبط سازد. ریمان قضیهٔ خود را براساس شش فرض بدون اثبات، ثابت کرد. همهٔ این فرضها بعداً اثبات شدند، به‌استثنای یکی از آنها که بعدها به‌فرضیهٔ ریمان شهرت یافت و شرح مجمل آن چنین است: تابع زتا را درنظر بگیرید

$$\zeta(s)=\sum_{n=1}^{\infty}\frac{1}{n^s}. \qquad (1)$$

این تابع نخستین بار در قرن هفدهم مطرح شد. ریاضیدانان بزرگی از قبیل برنولی‌ها و اویلر به‌آن علاقهٔ وافر نشان دادند. در قرن نوزدهم ریمان نیز آن را مورد بررسی قرار داد. هرچند سری $\sum_{n=1}^{\infty}1/n^s$ تنها به‌ازای $\mathrm{Re}\,s>1$ تعریف شده‌است، لکن به‌سادگی می‌توان ادامه‌ای تحلیلی برای آن یافت، یعنی تابعی تحلیلی چون f که بر تمام صفحهٔ مختلط تعریف شده است و تحدید آن در نیمصفحهٔ $\mathrm{Re}\,s>1$ با سری فوق برابر است. این تابع را هم با ζ نمایش می‌دهیم. همچنین ریمان نشان داد که معادلهٔ تابعی

$$\zeta(s)=2(2\pi)^{s-1}\Gamma(1-s)\sin\left(\frac{\pi s}{2}\right)\zeta(1-s) \qquad (2)$$

برقرار است. در اینجا Γ همان تابع گاما است با استفاده از (۱) و (۲) می‌توان نشان داد که کلیهٔ صفرهای تابع زتا، به استثنای صفرهای بدیهی (اعداد صحیح زوج منفی)، بر نوار $0<\mathrm{Re}\,s<1$ (موسوم به نوار بحرانی) واقع‌اند. فرضیهٔ ریمان می‌گوید که تمامی صفرهای غیربدیهی تابع زتا بر "خط بحرانی" $\mathrm{Re}\,s=1/2$ واقع‌اند. امروزه انتظار همگان آن است که درستی این فرضیه اثبات شود. نشان داده شده است که دست‌کم ۷۰ درصد صفرهای غیربدیهی (صفرهای روی نوار بحرانی) روی خط بحرانی واقع‌اند و نخستین ۷ میلیون صفر غیربدیهی نیز روی خط بحرانی قرار دارند. اما اثبات فرضیه بحث دیگری است. پس مسأله آن است که تحلیلی بودن تابع زتا را در خارج از نوار بحرانی مورد بررسی قرار دهیم. برای این کار می‌توان قطبهای ζ^{-1} را بررسی کرد. نخستین گام در این راه، محاسبهٔ $\zeta^{-1}(s)$ به‌ازای $\mathrm{Re}\,s>1$، بود.

فرض کنید μ تابع موبیوس باشد، یعنی

$$\mu(n)=\begin{cases}1 & n=1\\ (-1)^k & n=\prod_{i=1}^{k}p_i \quad (\text{ها اعداد اول متمایز هستند } p_i)\\ 0 & p^2|n \quad (\text{به‌ازای یك عدد اول } p)\end{cases}$$

به‌سادگی دیده می‌شود که $\sum_{n=1}^{\infty}\mu(n)/n^s$ نیز در ناحیهٔ $\mathrm{Re}\,s>1$ تحلیلی است و داریم

$$\left(\sum_{n=1}^{\infty}\frac{\mu(n)}{n^s}\right)\left(\sum_{n=1}^{\infty}\frac{1}{n^s}\right)=1,$$

یا

$$\frac{1}{\zeta(s)}=\sum_{n=1}^{\infty}\frac{\mu(n)}{n^s}.$$

سری فوق در $\mathrm{Re}\,s>1$ همگراست، پس تابع $1/\zeta$ در این ناحیه تحلیلی است. این مقدار اطلاع برای فرضیهٔ ریمان بسنده نیست (باید درجستجوی یك ادامهٔ تحلیلی تابع $\sum_{n=1}^{\infty}\mu(n)/n^s$ باشیم.) اگر $M(x)$ نمایانگر مجموعهای جزیی تابع موبیوس باشد:

$$M(x)=\sum_{n\leqslant x}\mu(n).$$

در این صورت با توجه به اینکه $M(x)$ روی هر بازهٔ به‌صورت $[n,n+1)$ ثابت است، داریم

$$\frac{1}{\zeta(s)}=\sum_{n=1}^{\infty}\frac{\mu(n)}{n^s}=\sum_{n=1}^{\infty}\frac{M(n)-M(n-1)}{n^s} \qquad (3)$$

$$=\sum_{n=1}^{\infty}M(n)\left[\frac{1}{n^s}-\frac{1}{(n-1)^s}\right]$$

$$=\sum_{n=1}^{\infty}M(n)\int_{n}^{n+1}\frac{s\,dx}{x^{s+1}}=s\sum_{n=1}^{\infty}\int_{n}^{n+1}\frac{M(x)\,dx}{x^{s+1}}$$

$$=s\int_{1}^{\infty}\frac{M(x)\,dx}{x^{s+1}}.$$

هرچند درستی رابطهٔ فوق را تنها به‌ازای $\mathrm{Re}\,s>1$ نشان دادیم، اما با توجه به‌اینکه تابعی که توسط انتگرال فوق تعریف می‌شود تحلیلی است، بنا برقضیه‌ای مقدماتی در آنالیز، تساوی فوق در هر ناحیه‌ای که این انتگرال همگرا باشد، برقرار است. اما با استفاده از معادلهٔ تابعی (۲)، کافی است ثابت کنیم که انتگرال فوق به‌ازای $\mathrm{Re}\,s>1/2$ همگراست. پس اگر انتگرال فوق را به‌صورت $s\int_1^\infty M(x)\,x^{-1/2}\,dx/x^{s+(1/2)}$ بیان کنیم، باید به‌بررسی رفتار $M(x)\,x^{-1/2}$ در همسایگی بینهایت بپردازیم. اگر این عبارت کراندار باشد، در آن صورت انتگرال فوق می‌تواند تابعی تحلیلی در $\sigma=\mathrm{Re}\,s>1/2$ تعریف کند و این، به تحلیلی بودن $1/\zeta(s)$ در $\sigma>1/2$ دلالت می‌کند و از این، درست‌بودن فرضیهٔ ریمان نتیجه می‌شود.

در سال ۱۸۹۷، مرتنس در مقاله‌ای در باب تابع زتا مقادیر عددی $M(n)$ و $\mu(n)$ را به‌ازای $n=1,2,\ldots,10^4$ محاسبه کرد و برمبنای این محاسبات حدس زد که نامساوی

$$|M(x)|<x^{1/2},\qquad x>1$$

درست است. امروزه این نامساوی را حدس مرتنس می‌نامند.

در سال ۱۹۶۳، نویباوئر[1] مقادیر $M(n)$ را برای $n\leqslant 7.8\times10^9$ محاسبه و مشاهده کرد که به‌ازای تمام این مقادیر

1. Neubauer

رابطهٔ فوق برقرار است. در واقع انتظار می‌رود که این نابرابری برای هر $n \leqslant 10^{20}$ درست باشد و حتی با توجه به قدرت کامپیوترهای امروزی نیز یافتن مثالی برای رد آن به‌طور مستقیم عملی نباشد [۷].

در نظریهٔ اعداد فرمولهای دقیقی وجود دارند که توابعی چون $\pi(x)$ را به صفرهای تابع زتا مربوط می‌سازند. در سال ۱۹۵۱ تیچمارش[1] با فرض درستی فرضیهٔ ریمان و ساده‌بودن صفرهای تابع زتا، فرمول مشابهی برای $M(x)$ کشف کرد. از این فرمول به‌سادگی نتیجه می‌شود که $\sum_\rho 1/|\rho\zeta'(\rho)|$ که سیگما روی صفرهای غیربدیهی تابع زتا $(\rho = 1/2 + i\gamma)$ است، واگراست. به‌علاوه اگر تعویض متغیر

$$x = e^y, \qquad -\infty < y < +\infty$$

را اعمال کنیم و قرار دهیم

$$h(y) = \lim_{k\to\infty} \sum_{\substack{\rho \\ |\gamma|<T_k}} \frac{e^{i\gamma y}}{\rho\zeta'(\rho)},$$

و

$$m(y) = M(x)x^{-1/2} = M(e^y)e^{-y/2},$$

آنگاه داریم

$$m(y) = h(y) + O(\min(1, e^{-y/2})).$$

(در تعریف h، T_k دنباله‌ای است که در فرمول تیچمارش ظاهر می‌شود و در نابرابری $k \leqslant T_k \leqslant k+1$ صدق می‌کند.) پس رفتار h و m در بینهایت یکسان است و مثلاً برای اثبات کراندار بودن $m(y)$ کافی است کراندار بودن $h(y)$ را ثابت کنیم. اما مشکلی که در مورد h وجود دارد این است که در مورد ضرایب $1/\rho\zeta'(\rho)$ اطلاع چندانی نداریم. این مشکل نیز در ۱۹۴۲ توسط اینگهام[2] رفع شد. پاسخ وی چنین بود که به‌جای مطالعهٔ مجموع فوق، سریهای متناهی از این نوع را بررسی کنیم. به‌عبارت دقیقتر، فرض کنید K تابعی نامنفی، زوج و از ردهٔ C^2 باشد و به علاوه شرط

$$K(y) = O((1+y^2)^{-1}), \quad y \to \infty$$

را برآورده سازد و اگر

$$k(t) = \int_{-\infty}^{+\infty} K(y)\, e^{-ity}\, dy,$$

و K به‌گونه‌ای باشد که $k(t)$ دارای محملی کراندار باشد (یعنی $k(t)$ خارج یک مجموعهٔ کراندار مثلاً $[-T, T]$ صفر باشد) و $K(0) = 1$، در این‌صورت پیچش[3] h و K یعنی h_K برابر خواهد بود با

$$h_K(y) = \int_{-\infty}^{+\infty} h(y-t)\, K(t)\, dt$$

$$= \sum_\rho \frac{e^{i\gamma y}}{\rho\zeta'(\rho)} \int_{-\infty}^{+\infty} K(t)e^{-i\gamma t}dt$$

$$= \sum_\rho k(\gamma)\frac{e^{i\gamma y}}{\rho\zeta'(\rho)}. \qquad (۴)$$

حال می‌توان نشان داد که به‌ازای هر y_0 داریم

$$\limsup_{y\to\infty} m(y) \geqslant h_K(y_0),$$

$$\liminf_{y\to\infty} m(y) \leqslant h_K(y_0).$$

پس برای رد حدس مرتنس کافی است K و y_0 را طوری انتخاب کنیم که داشته باشیم $|h_K(y_0)| > 1$. بنابراین لازم است $|h_K(y_0)|$ را بزرگ کنیم. فرض کنید $[-T, T]$ محمل $k(t)$ باشد. معمولاً T را برابر قسمت موهومی یکی از صفرهای غیربدیهی تابع زتا فرض می‌کنیم. به این ترتیب اگر T جزء موهومی این صفر باشد، داریم

$$h_K(y) = \sum_{|\gamma|<T} k(\gamma)\frac{e^{i\gamma y}}{\rho\zeta'(\rho)}$$

$$= 2\sum_{0<\gamma<T} k(\gamma)\frac{\cos(\gamma y - \psi_\gamma)}{|\rho\zeta'(\rho)|} \qquad (۵)$$

که در آن $\psi_\gamma = \operatorname{Arg}(\rho\zeta'(\rho))$. این تساوی آخر از اینجا به‌دست می‌آید که K و در نتیجه k توابع زوجی هستند.

در سال ۱۹۷۶، یورکات[1] و پیریموف[2] ملاحظه کردند که هرچند از لحاظ نظری در مورد اندازهٔ ضرایب $|\rho\zeta'(\rho)|^{-1}$ اطلاع چندانی نداریم، اما به‌طور عددی می‌دانیم که این ضرایب به‌سرعت کاهش می‌یابند بنابراین (۵) عمدتاً توسط چندجملهٔ اول معین می‌شود، پس می‌باید چندجملهٔ اول هم‌علامت و بزرگ باشند. به این منظور با توجه به اینکه $\sum_\rho 1/|\rho\zeta'(\rho)|$ واگراست، می‌باید K و y به‌گونه‌ای انتخاب شوند که برای چند صفر اول، $\cos(\gamma y - \psi_\gamma)$ و $k(\gamma)$ تقریباً برابر ۱ باشند. اما $\cos\theta$ زمانی تقریباً برابر ۱ است که θ تقریباً مضربی از 2π باشد. نزدیک‌کردن کمیت مثلثاتی مورد نظر به ۱، به این معناست که دستگاه معادلات دیوفانتی

$$|\gamma_j y - \psi_j - 2\pi m_j| < \varepsilon \qquad 1 \leqslant j \leqslant n,\ m_j \in \mathbb{Z} \qquad (۶)$$

را حل کنیم که در آن $\psi_j = \operatorname{Arg}((1/2+i\gamma_j)\zeta'(1/2+i\gamma_j))$ و γ_j قسمت موهومی jامین صفر تابع زتاست.

اما چگونه می‌توان اطمینان یافت که $k(\gamma)$ برای چند صفر اول تقریباً برابر ۱ است؟ برای این کار قرار می‌دهیم $k(t) = g(t/T)$ که در آن g تابعی است که خارج از $[-1, 1]$ صفر است. حال اگر g چنان باشد که در همسایگی صفر، $g(t)$ بسیار نزدیک ۱ باشد، وقتی T به‌اندازهٔ کافی بزرگ اختیار می‌شود، $k(\gamma)$ برای چند جملهٔ اول تقریباً برابر ۱ خواهد بود.

1. Titchmarsh 2. Ingham
3. convolution

1. Jurkat 2. Peyerimhoff

یورکات وپیریموف تابع gای به‌صورت زیرمعرفی کردند:

$$g(t)=\begin{cases}1-|t|\cos \pi t+\pi^{-1}\sin(\pi|t|), & |t|\leqslant 1\\ 0 & |t|\geqslant 1\end{cases}$$

و T را برابر قسمت موهومی ۵۳۶امین صفر تابـع زتـا (تقریباً $T=1000$) انتخاب کردند. همچنین روشی یافتند که دستگاه (۶) را به‌ازای εای که خیلی بزرگ نباشد حل می کرد و این‌الگوریتم را با استفاده ازیك‌کامپیوتر کوچك (جیبی) قابل‌برنامه‌نویسی برای این تابع به‌کار بردند و نتیجه گرفتند که

$$\limsup_{y\to\infty} m(y)\geqslant 0.779,\ \liminf_{y\to\infty} m(y)\leqslant -0.639.$$

اندکی بعد تی‌ریله[1] روش آنها را با یك‌کامپیوتر با سرعت زیادو برای ۱۵۰۰۰ صفر (به‌جای ۵۳۶ صفر) به‌کار برد و باصرف صدها ساعت وقت‌کامپیوتر نشان داد که

$$\limsup_{y\to\infty} m(y)\geqslant 0.860,\ \liminf_{y\to\infty} m(y)\leqslant -0.843.$$

با توجه به‌ارقام فوق به‌نظر می‌رسد که با تکنولوژی امروزی رد حدس مرتنس بااستفاده از الگوریتم یورکات-پیریموف امکان‌پذیر نباشد.

در سال ۱۹۸۵، ادلیز کو[2] و تی‌ریله با اصلاحاتی در روش یورکات-پیریموف موفق شدند نادرستی حدس مرتنس را به‌اثبات رسانند. اینان دستگاه معادلات دیوفانتی(۶)رانه برای nصفرنخست تابع زتا، بلکه برای آن n صفر تابع زتا کـه $1/(|\rho\zeta'(\rho)|)$ بیشترین مقدار را دارد در نظر گرفتند. به زبان دقیقتر، فرض کنید مثلاً نخستین ۴۰۰ صفر تابع زتـا را برحسب مقادیر $|\rho\zeta'(\rho)|$ بـه‌صورت صعـودی مرتب کنیـم: $\rho_1,\ldots,\rho_{400}$ که در آن $\rho_j=1/2+i\gamma_j$، و قراردهیم

$$\psi_j=\operatorname{Arg}(\rho_j\zeta'(\rho_j))=\operatorname{Arg}\left(\left(\tfrac{1}{2}+i\gamma_j\right)\zeta'\left(\tfrac{1}{2}+i\gamma_j\right)\right)$$

می‌خواهیم عدد حقیقی y و اعـداد صحیح m_j $(1\leqslant j\leqslant n)$ را به‌گونه‌ای بیابیم که کمیتهای $\eta_j=|\gamma_j y-\psi_j-2\pi m_j|$ همزمان کوچك باشند. برای یافتن چنین yای، آنهـا ازالگوریتم L^3 در شبکه‌ای مناسب‌استفاده کردند. قرارمی‌دهیم $\alpha_j=|\rho_j\zeta'(\rho_j)|^{-1/2}$ و فرض می کنیـم v یـك عـدد صحیح مثبت بـاشد. (معمـولاً $2n\leqslant v\leqslant 4n$) و شبکه‌ای راکه توسط بردارهای ستونی‌ماتریس $(n+2)\times(n+2)$ زیر تولید می‌شود، درنظر می گیریم:

$$\begin{matrix}
-[\alpha_1\psi_1 2^v] & [\alpha_1\gamma_1 2^{v-10}] & [2\pi\alpha_1 2^v] & 0 & 0\\
-[\alpha_2\psi_2 2^v] & [\alpha_2\gamma_2 2^{v-10}] & 0 & [2\pi\alpha_2 2^v] & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots\\
-[\alpha_n\psi_n 2^v] & [\alpha_n\gamma_n 2^{v-10}] & 0 & 0 & [2\pi\alpha_n 2^v]\\
2^v n^4 & 0 & 0 & 0 & 0\\
0 & 1 & 0 & 0 & 0
\end{matrix}$$

([x] به‌معنای بزرگترین عدد صحیح نابیشتر از x است.)

با استفاده از الگوریتم L^3 یك پایهٔ تحویل یافته برای شبکهٔ فوق‌به‌دست می‌آوریم: $[v_1^*,\ldots,v_{n+2}^*]$. از آنجاکه v_i^*ها پایه‌ای برای شبکـهٔ فوق تشکیل‌می‌دهند، درایهٔ $(n+1)$ام دست‌کم یکی از آنهـا مخالف صفراست. چون مقدار این درایه نسبت به‌سایر درایه‌ها نسبتاً بزرگ‌است، برای‌آنکه الگوریتم تحویل‌پایهٔ خوبی داشته باشیم(مجموعه‌ای از بردارهای با وزن کوچك فراهم شود)، لازم‌است‌که تنها یکی از بردارهای پایهٔ جدید مثلاً $\mathbf{w}$ دارای‌این خاصیت باشد. به‌علاوه ازآنجاکه در پایهٔ اولیه تنها یك بردار‌این خاصیت را دارد، درایـهٔ $(n+1)$ام $\mathbf{w}$ مضربی از $2^v n^4$ خواهد بود. بازهم برای اینکه الگوریتم تحویل پایهٔ خوبی داشته باشیم باید این مقدار $\pm 2^v n^4$ باشد. درتمام بررسیهایی‌که اینان به‌عمل آوردند (با تعویض مقادیر v و n) مشاهده‌کردندکه الگوریتم L^3 همواره هردوی این خواص را دارد. حال فرض کنید این درایهٔ $(n+1)$اُم ناصفردقیقاً $2^v n^4$ باشد، چون $\mathbf{w}$ برداری ازشبکه‌است، داریم

$$\exists z, m_1,\ldots,m_n\in\mathbf{Z},\ \mathbf{w}=v_1+zv_2-\sum_{k=1}^{n} m_k v_{k+2}.$$

پس درایهٔ jام $\mathbf{w}$ عبارت‌است از

$$z[\alpha_j\gamma_j 2^{v-10}]-[\alpha_j\psi_j 2^v]-m_j[2\pi\alpha_j 2^v].$$

برای اینکه طول $\mathbf{w}$ کوچك باشد، لازم‌است‌که قدرمطلق هریك‌از درایه‌هایش کوچك‌باشد. پس‌هریك ازعبارات فوق نیز بایدکوچك باشد

$$z\alpha_j\gamma_j 2^{v-10}-\alpha_j\psi_j 2^v-m_j 2\pi\alpha_j 2^v.$$

و در نتیجه اگر $y=z/2^{10}$، $|\alpha_j|(\gamma_j y-\psi_j-2\pi m_j)$ بسیار کوچك خواهد شد. بررسیهای عملی نشان دادکـه الگوریتم L^3، در واقع این خاصیت را دارد و دستگاه دیوفانتی(۶) را به‌ازای ε کوچك حل می کند.

دراینجا ذکر دونکته ضروری‌است. نخست آنکه علت حضور α_jها در پایه این‌است که هدف، بزرگ کردن

$$\sum_{j=1}^{n}\alpha_j^2\cos(\gamma_j y-\psi_j-2\pi m_j)$$

بود. اما اگر$(\gamma_j y-\psi_j-2\pi m_j)$هاهمگی کوچك باشند،عبارت فوق تقریباً برابر

$$\sum_{j=1}^{n}\alpha_j^2-\frac{1}{2}\sum_{j=1}^{n}\alpha_j^2(\gamma_j y-\psi_j-2\pi m_j)^2$$

است. پس باید $|\alpha_j(\gamma_j y-\psi_j-2\pi m_j)|$ راکوچك‌کنیم. واین درحقیقت‌متناظراست باکمینه کردن نرم‌اقلیدسی بردار$(\beta_1,\ldots,\beta_n)$، کـه درواقع به‌وسیلهٔ الگوریتم L^3 انجام می‌پذیرد. و دوم آنکه اگر بخواهیم مقادیری از y را بیابیم که $h_K(y)<-1$، کافی است درشبکهٔ فوق ψ_j را بـه $\psi_j+\pi$ تبدیل کنیم. و بدین ترتیب ادلیز کو و تی‌ریلهٔ با استفاده از الگوریتم L^3، عدم صحت حدس مرتنس را نشان دادند.

بررسیهای عددی نشان می‌دهد که با همان ترتیبی کـه برای

1. te Riele 2. Odlyzko

نخستین ۴۰۰ صفر تابع زتا قائل شدیم مقدار $\left|\sum_{j=1}^{n} |\rho_j \zeta'(\rho_j)|\right|$، به‌ازای $n = ۵۴$ از ۱ تجاوز می‌کند و به ازای $n = ۷۰$ تقریباً برابر ۱٫۰۷۸۷ می‌شود. بنابراین، کافی بود دستگاه معادلات دیوفانتی را برای $n = ۷۰$ حل کنند. به‌علاوه برای آنکه عامل $k(\gamma)$ نسبتاً بزرگ باشد، T را برابر جزء موهومی ۲۰۰۰امین صفر تابع زتا، یعنی تقریباً معادل ۲۵۱۵ تعریف کردند. همچنین بررسیهای اینان نشان داد که اگر صفرهای تابع زتا به‌طور تصادفی روی خط بحرانی ($\mathrm{Re}\, z = ۱/۲$) توزیع شده باشند، مقدار y که دستگاه (۶) را حل می‌کند تقریباً درحدود $۱۰^{۷۰}$ خواهد بود. پس خطای کوچکی در محاسبهٔ γ_jها به‌خطای بزرگی منجر می‌شد. به‌این علت ادلیزکو و تی‌ریله ۲۰۰۰ صفر اول تابع‌زتا را با استفاده‌از فرمول اویلر-مکلورن تا صد رقم اعشار محاسبه کردند. و به‌ازای مقادیر مختلف n و v دستگاه معادلات دیوفانتی (۶) را حل کردند و مقدار z را محاسبه کردند. سپس به y مقادیری نزدیک $z/۱۰۲۴$ دادند و به‌نتایج زیر رسیدند:

$$n = ۷۰,\ v = ۲۳۰$$

$$y = -۱۴۰۴۵۲۸۹۶۸۰۵۹۲۹۹۸۰۴۶۷۹۰۳۶۱۶۳۰۳۹۹۷ \\ ۸۱۱۲۷۴۰۰۵۹۱۹۹۹۷۸۹۷۳۸۰۳۹۹۶۵۹۶۰۷۶۴ \\ ۲٫۵۲۱۵۰۵$$

$$h_K(y) = ۱٫۰۶۱۵۴۵$$

$$n = ۷۰,\ v = ۲۳۰$$

$$y = ۳۲۰۹۷۰۲۵۷۷۲۹۲۲۶۵۵۸۶۹۷۴۰۰۰۰۱۸۶۲۱۱ \\ ۳۰۷۰۹۹۷۹۷۱۴۴۵۴۰۳۴۹۰۶۲۶۸۲۸۰۵۳۲۱۶۵ \\ ۱٫۶۹۷۴۱۹$$

$$h_K(y) = -۱٫۰۰۹۷۴۹$$

مرجع اصلی این بخش [۷] ومراجع فرعی آن [۸]، [۹]، [۱۰]، و [۱۲] است.

۵. کاربردی دیگر

درکارهای عملی گاهی با مسائلی مواجه می‌شویم که هیچ الگوریتم با زمان چندجمله‌ای برای حل آنها موجود نیست. چنین مسائلی‌را NP-تمام می‌نامیم. مثلاً مسألهٔ یافتن تمام جایگشتهای روی n حرف، از زمرهٔ این مسائل است زیرا $n!$ جایگشت وجود دارد. البته در موارد دیگر اثبات NP-تمام بودن از چنین وضوحی برخوردار نیست. یکی دیگر از مسائلی که NP-تمام بودن آن مشخص شده است و در واقع یکی از ساده‌ترین این‌نوع مسائل است، مسألهٔ مجموع زیرمجموعه‌ای (مسألهٔ کوله‌پشتی‌دقیق)[1] است. صورت‌این مسأله این‌است: فرض کنید اعداد صحیح مثبت a_1، ...، a_n، و b داده‌شده‌اند. زیرمجموعهٔ J از $\{۱, \ldots, n\}$ را بیابید به‌طوری که $\sum_{i \in J} a_i = b$. به‌عبارت دیگر مسألهٔ برنامه‌ریزی با اعداد صحیح

$$\sum a_i x_i = M, \qquad x_i \in \{۰, ۱\} \qquad (*)$$

را‌حل کنید. NP-تمام بودن این‌مسأله سبب شده‌است که‌کاربردهای وسیعی در رمزنگاری داشته باشد.

هرچند هیچ الگوریتم با زمان چندجمله‌ای برای‌حل این‌مسأله موجود نیست، لکن رده‌های خاصی از این نوع مسائل‌را می‌توان با الگوریتمهای با زمان چندجمله‌ای حل کرد. از جملهٔ آنهامسائل کوله‌پشتی با چگالی کوچك می‌باشد. چگالی یك مسألهٔ مجموع زیرمجموعه‌ای به‌صورت زیرتعریف می‌شود:

$$d(a) = \frac{n}{\log_۲(\max\{a_i \mid ۱ \leqslant i \leqslant n\})},\ a = (a_1, \ldots, a_n).$$

درسال ۱۹۸۵ ادلیزکو ولاگاریس [۳] برمبنای الگوریتم L^3 روش ساده‌ای عرضه کردندکه برخی از این مسائل را حل کرد. آنها ثابت‌کردند که برای هر n داده شده، یك عدد ثابت $d_c(n)$ وجود داردکه هرمسألهٔ مجموع زیرمجموعه‌ای با $d(a) \leqslant d_c(n)$ را می‌توان با این‌روش‌حل کرد. روش آنها برای حل (*) بدین‌قرار است:

شبکهٔ تولید شده توسط بردارهای سطری ماتریس $(n+1) \times (n+1)$ زیررا درنظربگیرید

$$B = \begin{bmatrix} ۱ & & \bigcirc & -a_1 \\ & ۱ & & -a_2 \\ & & \ddots & \vdots \\ \bigcirc & & ۱ & -a_n \\ & & & M \end{bmatrix}$$

هر بردار دراین شبکه به‌صورت

$$\left(y_1, y_2, \ldots, y_n, M - \sum_{i=1}^{n} a_i y_i\right)$$

است. حال‌اگر $(x_1, \ldots, x_n)$ جوابی برای‌مسألهٔ (*) باشد، دراین صورت $(x_1, \ldots, x_n, ۰)$ در شبکهٔ فوق قرار خواهد داشت و بردارکوچکی ازشبکه‌است. پس به‌این‌ترتیب عمل‌می‌کنیم: نخست یك پایهٔ تحویل یافته $B = [b_1, \ldots, b_{n+1}]$ برای شبکه به‌دست می‌آوریم که درآن $b_i = (b_{i1}, b_{i2}, \ldots, b_{in+1})$. حال‌اگر یکی از بردارهای این‌پایه مثلاً b_i، برداری با مؤلفه‌های ۰ و ۱ و −۱ باشد، قرار می‌دهیم $x_j = |b_{ij}|$ و بررسی می‌کنیم که بردار $\mathbf{x}$ یك جواب برای مسألهٔ (*) هست یا خیر. اگرجواب منفی‌بود، درآن صورت M را با $M' = \sum_{i=1}^{n} a_i - M$ عوض می‌کنیم و الگوریتم را تکرار می‌کنیم.

این الگوریتم در حالت‌کلی لزوماً به‌جوابی نمی‌انجامد اما، همان‌طورکه گفته شد، برای هر مسألهٔ با $d(a) \leqslant d_c(n)$ حتماً به‌جواب می‌رسد.

بعدها، با اعمال چند الگوریتم کمکی، ثابت d_c اصلاح شد[۴].

1. exact knapsack problem

۶. حرف آخر

لنسترا الگوریتم L^3 را تعمیم داده و در مورد مسائل متنوعی به‌کار برده است: هیأت اعداد جبری، چندجمله‌ایهای چندمتغیره روی هیأتهای متناهی، وهیأتهای با مشخصهٔ p و $\mathbf{Q}$.

تدوین‌کنندگان این مقاله نیز این الگوریتم را در مورد طرحهای بلوکی به‌کار گرفته‌اند و نتایج تحقیقات خود را درمقالهٔ جداگانه‌ای منتشر خواهند کرد.

مراجع

1. D. E. Knuth, *The Art of Computer Programming*, Vol. 2. *Seminumerical Algorithms*, Addison-Wesley (1981).
2. D. L. Kreher and S. P. Radziszowski, "Solving subset sum problems with the L^3 algorithm," *J. Combin. Math. and Computing*, to appear.
3. J. C. Lagarias and A. M. Odlyzko, "Solving low-density subset sum problem," *J. ACM.*, (1) **32** (1985) 229-246.
4. S. Landau, "Factoring polynomials quickly," *Notices*, (1) **34** (1987) 3-8.
5. A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász, "Factoring polynomials with rational coefficients," *Math. Ann.*, **261** (1982) 515-534.
6. R. Lidl, *Finite Fields*, Cambridge Univ. Press (1987).
7. A. M. Odlyzko and H. J. J. te Riele, "Disproof of the Mertens conjecture," *J. für die Reine und Angewandte Mathematik*, **357** (1985) 138-160.

۸. تام آپوستل، *نظریهٔ تحلیلی اعداد*، ترجمهٔ علی‌اکبر عالم‌زاده، و علی‌اکبر رحیم‌زادهٔ ثانی، انتشارات شباهنگ، چاپ اول، تابستان ۶۷.

۹. ریموند ایوب، «اویلر و تابع زتا»، ترجمهٔ شاهین آجودانی نمینی، *جنگ ریاضی دانشجو*، جلد اوّل (۱۳۶۶)، ۴۳ـ۶۷.

۱۰. دان زاگیر، «نخستین ۵۰ میلیون عدد اوّل»، ترجمهٔ رحیم زارع نهندی، *نشر ریاضی*، سال ۱، شمارهٔ ۳ (۱۳۶۷) ۱۸۹ـ۱۹۹.

۱۱. سعید عباس بندی، *الگوریتم L^3*، پروژهٔ دانشجویی، دانشکدهٔ علوم، دانشگاه تهران، (۱۳۶۸).

۱۲. هربرت ویلف، «نگاهی به فرضیهٔ ریمان»، ترجمهٔ مجتبی منیری، *نشر ریاضی*، سال ۱، شمارهٔ ۱ (۱۳۶۷)، ۱۶ـ۱۸.

◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆

* غلامرضا برادران خسروشاهی، شاهین آجودانی نمینی، و محمد رجبی طرخورانی، دانشگاه تهران