Parametrisierung von Pollards Rho-Methode

Finn Rudolph
21.01.2024

Erarbeitungsort: Hauptstraße 28, 96178 Pommersfelden
Fachgebiet: Mathematik / Informatik
Wettbewerbssparte: Jugend forscht
Bundesland: Bayern
Wettbewerbsjahr: 2024

Projektüberblick

Pollards Rho-Methode ist einer der schnellsten Algorithmen zur Faktorisierung kleiner Zahlen. Bei der Implementierung des Algorithmus kann ein Parameter k gewählt werden, der unter Umständen großen Einfluss auf die Laufzeit des Algorithmus hat, sowohl im positiven als auch im negativen Sinn. In dieser Arbeit soll untersucht werden, wie k bestmöglich gewählt wird. Insbesondere ist der Fall interessant, wenn der Algorithmus auf mehreren Maschinen parallel ausgeführt wird, weil dann für jede Maschine k separat gewählt werden kann. Für den Fall einer und zweier Maschinen konnten theoretische Ergebnisse erzielt werden, im Fall zweier Maschinen bleiben aber noch Fragen offen. Diese Ergebnisse decken sich mit durchgeführten Experimenten. Offen bleibt auch die Frage der optimalen Parametrisierung für drei oder mehr Maschinen.

Inhaltsverzeichnis

1 Zusammenfassung

In dieser Arbeit wird die Frage behandelt, wie der Parameter k in Pollards Rho-Methode optimal gewählt werden kann. Ein größerer Wert von k erhöht grundsätzlich die Laufzeit, kann aber auch zu einer deutlichen Verringerung führen, wenn die Primfaktoren der zu faktorisierenden Zahl günstige Eigenschaften haben. Im Allgemeinen ist über die Primfaktoren allerdings nichts bekannt, sie sollen ja durch den Algorithmus bestimmt werden. Daher stellt sich die Frage, welcher Wert von k im Mittel am besten ist. Es ergibt sich insbesondere dann ein interessantes Problem, wenn der Algorithmus auf mehreren Maschinen parallelisiert wird, weil k dann für jede Maschine gewählt werden muss. Unter weithin anerkannten Annahmen über Pollards Rho-Algorithmus konnten grundlegende Methoden zur Beantwortung dieser Frage entwickelt werden. Mit diesen war es möglich zu zeigen, dass $k = 1$ für eine Maschine optimal ist. Im Fall zweier Maschinen wird gezeigt, dass $k_1 = k_2 = 1$ besser ist als wenn k_1 und k_2 Primzahlen sind oder wenn $k_1 = k_2 > 1$ gilt. k_i bezeichnet den Wert von k für die i-te Maschine. Anschließend werden Laufzeitmessungen vorgestellt, die die theoretischen Ergebnisse bestätigen.

2 Motivation und Fragestellung

Pollards Rho-Methode bleibt trotz der Existenz asymptotisch schnellerer Algorithmen einer der meist verwendeten Algorithmen zur Faktorisierung kleiner Zahlen. Gleichzeitig werden Leistungssteigerungen bei modernen Computern häufig durch größere Nebenläufigkeit (z. B. mehr Prozessorkerne) erzielt. Um die Rho-Methode schnellstmöglich zu implementieren, ist es also nötig zu untersuchen, wie sie am besten parallel ausgeführt werden kann. Bevor die Fragestellung präzise formuliert werden kann, soll jedoch Pollards Rho-Methode vorgestellt werden.

2.1 Pollards Rho-Methode

Sei n die zu faktorisierende Zahl. Es wird angenommen, dass n ungerade und keine Potenz einer natürlichen Zahl ist, da sonst einfach ein Faktor gefunden werden kann. Sei $h : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ mit $h : x \mapsto x^{2k} + 1$ für einen Parameter $1 \leq k \in \mathbb{N}$. Man wähle einen zufälligen Anfangswert $x_0 \in \mathbb{Z}/n\mathbb{Z}$ und betrachte die Folge $(x_i)_{i\in\mathbb{N}}$, definiert durch $x_i = h(x_{i-1})$. Da $(x_i)_{i\in\mathbb{N}}$ über der endlichen Menge $\mathbb{Z}/n\mathbb{Z}$ definiert ist, ist die Folge ab einem bestimmten Punkt periodisch. Sei p ein Primfaktor von n und $\pi : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/p\mathbb{Z}$ die natürliche Projektion. Die Idee der Rho-Methode ist, zwei Folgenglieder $x_i, x_j \in \mathbb{Z}/n\mathbb{Z}$ zu finden, sodass $x_i \neq x_j$ aber $\pi(x_i) = \pi(x_j)$. Dann ist nämlich $\gcd(n, x_i - x_j)$ ein echter Faktor von n. Das Ereignis, dass eine Folge einen Wert zweimal annimmt, wird eine *Kollision* in dieser Folge genannt. Nimmt man an, dass die Periodenlänge von $(x_i)_{i\in\mathbb{N}}$ in $\mathbb{Z}/n\mathbb{Z}$ deutlich länger als die Periodenlänge in $\mathbb{Z}/p\mathbb{Z}$ ist, reicht es aus, x_i, x_j mit $i \neq j$ zu finden, die kongruent modulo p sind. Die Annahme ist plausibel, weil für den kleinsten Primfaktor $p \leq \sqrt{n}$ gilt, es also deutlich weniger mögliche Werte für $\pi(x_i)$ als x_i gibt. Im Folgenden ist p immer der kleinste Primfaktor von n. Außerdem wird angenommen, dass die anderen Primfaktoren von n so viel größer als p sind, dass die Wahrscheinlichkeit einer Kollision modulo eines anderen Primfaktors vernachlässigbar gering ist. Zum Finden solcher x_i, x_j ist es hilfreich, den funktionalen Graphen von h zu betrachten.

Definition 1 (Funktionaler Graph)**.** *Sei X eine endliche Menge und $f : X \to X$ eine Abbildung.*

Der funktionale Graph von f, geschrieben $\gamma(f)$, ist der gerichtete Graph mit Knotenmenge X und Kantenmenge E, wobei die Kante $(x, y) \in X \times X$ genau dann in E liegt, wenn $f(x) = y$.

Es ist leicht zu zeigen, dass jede Zusammenhangskomponente eines funktionalen Graphen aus einem Zyklus und an den Zyklusknoten gewurzelten Bäumen besteht. In $\gamma(h)$ ist x_0 also ein Knoten in einem der Bäume, der durch seinen Pfad zur Wurzel mit dem Zyklus seiner Zusammenhangskomponente verbunden ist. Die Folge $(x_i)_{i\in\mathbb{N}}$ startet bei x_0 und „läuft“ durch den Graphen, wobei immer die eindeutige von einem Knoten ausgehende Kante entlanggegangen wird. An der Wurzel des Baums von x_0 wird der Zyklus in der Zusammenhangskomponente von x_0 betreten, und ab genau diesem Punkt ist $(x_i)_{i\in\mathbb{N}}$ periodisch. Bei Auftritt der ersten Kollision bildet der von $(x_i)_{i\in\mathbb{N}}$ abgelaufene Pfad die Form eines „ρ“ in $\gamma(h)$.

Sei f die Abbildung h, betrachtet in $\mathbb{Z}/p\mathbb{Z}$. Die Folge $(\pi(x_i))_{i\in\mathbb{N}}$ „läuft“ also durch $\gamma(f)$, beginnend von $\pi(x_0)$. Unser Ziel, das Finden einer Kollision von $(\pi(x_i))_{i\in\mathbb{N}}$, ist also äquivalent dazu, einen Knoten in $\gamma(f)$ zu finden, der zweimal in dem von $(\pi(x_i))_{i\in\mathbb{N}}$ abgelaufenen Pfad erscheint. Dafür kann Floyds Algorithmus zur Zykluserkennung in $\gamma(f)$ verwendet werden. Floyds Algorithmus macht sich zunutze, dass es ein $1 \leq r \in \mathbb{N}$ mit $\pi(x_r) = \pi(x_{2r})$ geben muss (Knuth, 1998, S. 7). Sei $\mu(f, \pi(x_0))$ die Höhe von $\pi(x_0)$ in seinem Baum und $\lambda(f, \pi(x_0))$ die Länge des Zyklus von $\pi(x_0)$. Für das minimale solcher r gilt dann $r \leq \mu(f, \pi(x_0)) + \lambda(f, \pi(x_0))$ (Knuth, 1998, S. 7). Wir nennen $\nu(f, \pi(x_0)) = \mu(f, \pi(x_0)) + \lambda(f, \pi(x_0))$ die Rho-Länge von $\pi(x_0)$ in f. Es würde also genügen, die Folgen $(\pi(x_i))_{i\in\mathbb{N}}$ und $(\pi(x_{2i}))_{n\in\mathbb{N}}$ gleichzeitig Glied für Glied zu berechnen und in jedem Schritt zu überprüfen, ob $\pi(x_i) = \pi(x_{2i})$. Das kann aber nicht explizit geschehen, da p unbekannt ist. Stattdessen werden $(x_i)_{i\in\mathbb{N}}$ und $(x_{2i})_{i\in\mathbb{N}}$ Glied für Glied berechnet. Das Überprüfen, ob $\pi(x_i) = \pi(x_{2i})$ geschieht durch Berechnung von $\gcd(n, x_i - x_{2i})$. So erhält man nach maximal $\nu(f, \pi(x_0))$ Schritten die gewünschte Kollision. Die Methode kann wie folgt zusammengefasst werden.

Algorithmus : Pollards Rho-Methode

```
x ← zufällige natürliche Zahl zwischen 0 und n − 1
y ← x
while TRUE do
    x ← x^{2k} + 1  mod n
    y ← (y^{2k} + 1)^{2k} + 1  mod n
    g ← gcd(n, x − y)
    if g ≠ 1 and g ≠ n then
        return g
    end
end
```

Die Analyse von Pollards Rho-Algorithmus erweist sich als schwierig, es ist bis dato keine rigorose Laufzeitanalyse bekannt. Unter heuristischen Annahmen lässt sich die Laufzeit allerdings gut abschätzen. Als erste Vereinfachung wird statt der mittleren Anzahl an Iterationen von Floyds Algorithmus die mittlere Rho-Länge analysiert. Eine zentrale Annahme dreht sich um die Verteilung der Rho-Längen in $\gamma(f)$, für deren Formulierung der Begriff einer asymptotischen Näherung benötigt wird.

Definition 2 (Asymptotische Näherung)**.** *Eine Funktion $f : \mathbb{R} \to \mathbb{R}$ heißt genau dann asym-*

ptotische Näherung von einer Funktion $g : \mathbb{R} \to \mathbb{R}$, oder asymptotisch zu g, wenn

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 1$$

In diesem Fall schreiben wir $f \sim g$.

Sei $A(n)$ die Menge der Abbildungen $\mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ für $n \in \mathbb{N}$. Über die Verteilung der Rho-Längen wird folgende Annahme getroffen, die auch *Random Mapping Assumption* (RMA) genannt wird.

Annahme (Random Mapping Assumption)**.** *Sei $f : \mathbb{Z}/p\mathbb{Z} \to \mathbb{Z}/p\mathbb{Z}$ mit $f : x \mapsto x^{2k} + 1$ und $d = \gcd(p-1, 2k)$. Seien $x_0 \in \mathbb{Z}/p\mathbb{Z}$ und $y_0 \in \mathbb{Z}/((p-1)/d)\mathbb{Z}$ zufällig und $g \in A((p-1)/d)$ zufällig. Dann gilt $\mathbb{P}(\nu(f, x_0) = m) \sim \mathbb{P}(\nu(g, y_0) = m)$ für $p \to \infty$.*

In anderen Worten sagt die Random Mapping Assumption, dass sich die Verteilung der Rho-Längen von $f : x \mapsto x^{2k} + 1$ wie bei einer zufälligen Funktion aus $A((p-1)/d)$ verhält. Insbesondere verhält sich $x \mapsto x^2 + 1$ bezüglich der Rho-Längen wie ein zufällige Funktion $\mathbb{Z}/(p-1)\mathbb{Z} \to \mathbb{Z}/(p-1)\mathbb{Z}$. Brent und Pollard, 1981 geben eine Begründung für RMA. Im Folgenden wird statt $(p-1)/d$ einfach p/d verwendet, da $p \sim p-1$ für $p \to \infty$.

Für $k = 1$ ist unter RMA die erwartete Anzahl an Iterationen der while-Schleife asymptotisch zu $\sqrt{\pi p/2}$ (Knuth, 1998, S. 8). Da die Berechnung des größten gemeinsamen Teilers $O(\ln n)$ Schritte benötigt, ist die erwartete Laufzeit des Algorithmus $O(\sqrt{p} \ln n)$. Durch eine einfache Modifikation kann die Dauer des gcd amortisiert werden, sodass sich die Laufzeit auf $O(\sqrt{p})$ verringert (Brent, 1980). Damit ist pro Iteration also nur noch die Zeit zur Berechnung der $2k$-ten Potenzen von x und y relevant, was durchschnittlich in $c \lg 2k$ Schritten möglich ist, wobei c eine hier unwichtige Konstante ist. Mit $\lg x$ wird der Logarithmus zur Basis 2 bezeichnet.

2.2 Parallelisierung der Rho-Methode

Sei M die Anzahl verfügbarer Maschinen. Eine *Maschine* meint hier nicht zwingend einen Computer, sondern eine Ressource, auf der ein sequentielles Programm ausgeführt werden kann, was beispielsweise auch ein Prozessorthread sein kann. Die Rho-Methode lässt sich parallelisieren, indem M Anfangswerte zufällig und unabhängig voneinander gewählt werden und der Algorithmus auf jeder der M Maschinen ausgeführt wird, bis eine der Maschinen einen Faktor findet. Nun ergibt sich folgende Frage, die in dieser Arbeit behandelt werden soll: *Wie wählt man den Parameter k für jede Maschine optimal, um eine möglichst geringe Laufzeit zu erzielen?* Das ist nicht sofort klar, da durch ein größeres k möglicherweise $\gcd(p-1, 2k)$ groß ist, sodass die Zahl an Iterationen um einen Faktor $\sqrt{\gcd(p-1, 2k) - 1}$ sinkt. Allerdings steigt die Dauer einer Iteration um einen Faktor $\lg 2k$. Da es sich bei den Veränderungen in der Laufzeit durch Veränderung von k um konstante Faktoren handelt, wird für den Vergleich der Laufzeit nicht die O-Notation verwendet, sondern eine asymptotische Näherung für die erwartete Zahl an Zeiteinheiten bestimmt, wenn $p \to \infty$. Wir definieren eine *Zeiteinheit* als die Dauer einer Iteration für $k = 1$, bei einer Maschine mit Parameter k dauert eine Iteration also $\lg 2k$ Zeiteinheiten. Im Gegensatz zur O-Notation kann zwischen zwei Funktionen, die asymptotisch zueinander sind, für große n kein konstanter Faktor liegen, sodass sich Veränderungen um konstante Faktoren sinnvoll vergleichen lassen.

Sei $k_1, \ldots, k_M, 1 \leq k_i \in \mathbb{N}$ eine Zuordnung von k-Werten für M Maschinen. Mit $L_{k_1,\ldots,k_M}(p)$ wird die erwartete Laufzeit des parallelen Rho-Algorithmus mit entsprechenden k-Werten bezeichnet. Sei $h_i = p/(\gcd(p-1, 2k_i) - 1)$. Unter RMA gilt

$$L_{k_1,\ldots,k_M}(p) = \mathbb{E}\left(\min_{i=1}^{M} X_i\right) \tag{1}$$

wobei X_i die gleichverteilte Zufallsvariable über $A(h_i) \times \mathbb{Z}/h_i\mathbb{Z}$ ist, mit $X_i(f, x_0) = \nu(f, x_0)$ für $f \in A(h_i), x_0 \in \mathbb{Z}/h_i\mathbb{Z}$. In $L_{k_1,\ldots,k_M}(p)$ ist $\gcd(p-1, 2k_i)$ für alle $1 \leq i \leq M$ noch fixiert, der Erwartungswert über alle Möglichkeiten von $\gcd(p-1, 2k_i)$ wird erst in Abschnitt 5 behandelt. Eine Schwierigkeit in der Herleitung einer Formel für $L_{k_1,\ldots,k_M}(p)$ ist, dass X_i und X_j nicht unabhängig sind, wenn $k_i = k_j$. Grund dafür ist, dass f in diesem Fall gleich ist, sodass sich die i-te und j-te Maschine im gleichen funktionalen Graphen bewegen. Daher wird in Abschnitt 3 der Fall unabhängiger Maschinen behandelt, und in Abschnitt 4 der Fall abhängiger Maschinen für $M = 2$.

3 Eine Formel im Fall unabhängiger Maschinen

In diesem Abschnitt wird eine Formel für $L_{k_1,\ldots,k_M}(p)$ hergeleitet, die im Fall paarweise verschiedener k-Werte gilt. Die Rho-Längen verschiedener Maschinen sind hier stochastisch unabhängig. Sei $h_i = p/(\gcd(p-1, 2k_i) - 1)$ und s_i die Anzahl an Iterationen, nach denen bei der i-ten Maschine erstmals eine Kollision auftritt. Sei $t_i = s_i \lg 2k_i$ die Zeit, nach der bei Maschine i erstmals eine Kollision auftritt und $t_{\min} = \min_{i=1}^{M} t_i$. Das Ziel ist die Bestimmung von $\mathbb{E}(t_{\min})$.

Eine Formel für $\mathbb{P}(s_i > s)$. Nach RMA gilt $\mathbb{P}(s_i = s) = \mathbb{P}(\nu(f, x_0) = s)$ für eine zufällig gewählte Funktion $f \in A(h_i)$ und ein zufälliges $x_0 \in \mathbb{Z}/h_i\mathbb{Z}$. Für eine zufällige Funktion $f \in A(h_i)$ ist die Wahrscheinlichkeit einer Kollision im j-ten Schritt j/h_i, wenn in den ersten $j-1$ Schritten keine Kollision aufgetreten ist, da jeder der h_i möglichen Werte gleich wahrscheinlich ist und j von ihnen zu einer Kollision führen. Es gilt also

$$\mathbb{P}(s_i > s) = \mathbb{P}(\nu(f, x_0) > s) = \prod_{j=0}^{s} \left(1 - \frac{j}{h_i}\right)$$

Da die i-te Maschine in t Zeiteinheiten $\lfloor t/\lg 2k_i \rfloor$ Schritte ausführt, gilt $\mathbb{P}(t_i > t) \sim \mathbb{P}(s_i > t/\lg 2k_i)$ für große t. Das Weglassen der Gaußklammern lässt sich damit rechtfertigen, dass für ausreichend großes p nur große t relevant für den Erwartungswert von $t_{\min}$ sind, bei denen es kaum einen Unterschied macht. Zur Herleitung der Formel werden noch weitere solcher Vereinfachungen nötig sein, diese zu beweisen ist aber meist uninteressant und aufwändig. Im Abschnitt *Weglassen von* $O(j^2/h_i^2)$ wird für eine von ihnen genauer begründet, um zu zeigen, wie ein solches Argument aussehen kann. Durch Anwenden der Restgliedabschätzung $\exp x = 1 + x + O(x^2)$, wenn $|x| < 1$, auf $x = -j/h_i$ gilt

$$\mathbb{P}(s_i > s) = \prod_{j=0}^{s} \left(\exp\left(\frac{-j}{h_i}\right) - O\left(\frac{j^2}{h_i^2}\right)\right)$$

$\mathbb{E}(t_{\min})$ lässt sich nun wie folgt ausdrücken, wobei $t_{\max} = p \max_{i=0}^{M} \lg 2k_i$ die maximal mögliche

Anzahl an Zeiteinheiten ist.

$$\begin{aligned}\mathbb{E}(t_{\min}) &= \sum_{t=0}^{t_{\max}} t\left(\prod_{i=1}^{M}\mathbb{P}(t_i > t-1)\right)\mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i)\\ &\sim \sum_{t=0}^{t_{\max}} t\left(\prod_{i=1}^{M}\prod_{j=0}^{(t-1)/\lg 2k_i}\left(\exp\left(\frac{-j}{h_i}\right) - O\left(\frac{j^2}{h_i^2}\right)\right)\right)\mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i)\end{aligned}$$

Der erste Faktor ist die Wahrscheinlichkeit, dass vor Zeitpunkt t keine Kollision aufgetreten ist. Der zweite Faktor ist die Wahrscheinlichkeit, dass bei Zeitpunkt t mindestens eine Kollision auftritt. $\mathbb{P}_{t_i>t-1\,\forall i}(\cdots)$ ist die Wahrscheinlichkeit dass $t_i = t$ für mindestens ein i, gegeben $t_i > t-1$ für alle i. Bevor diese Wahrscheinlichkeit bestimmt wird, soll gezeigt werden, dass der erste Faktor stark vereinfacht werden kann.

Weglassen von $O(j^2/h_i^2)$. Intuitiv ist es aus folgendem Grund gerechtfertigt, $O(j^2/h_i^2)$ in obiger Formel wegzulassen. Wenn $s = (t-1)/\lg 2k_i$ deutlich kleiner als h_i ist, ist auch j deutlich kleiner als h_i und damit j^2/h_i^2 nahe 0. Wenn dagegen s nahe h_i ist, ist $\mathbb{P}(s_i > s)$ sehr klein, sodass der Beitrag zum Erwartungswert vernachlässigbar ist. Präzise lässt sich das wie folgt begründen. Zunächst wird $s \le h_i^{3/5}$ angenommen und gezeigt, dass dann $\prod_{j=0}^{(t-1)/\lg 2k_i}(\exp(-j/h_i) - O(j^2/h_i^2)) \sim \prod_{j=0}^{(t-1)/\lg 2k_i}\exp(-j/h_i)$. Durch Ausmultiplizieren des Produkts erhält man

$$\begin{aligned}&\left|\prod_{j=0}^{s}\left(\exp\left(\frac{-j}{h_i}\right) - O\left(\frac{j^2}{h_i^2}\right)\right) - \prod_{j=0}^{s}\exp\left(\frac{-j}{h_i}\right)\right|\\ &= \left|-\sum_{0\le a\le s} O\left(\frac{a^2}{h_i^2}\right)\prod_{j=0,j\notin\{a\}}^{s}\exp\left(\frac{-j}{h_i}\right) + \sum_{0\le a<b\le s} O\left(\frac{a^2b^2}{h_i^4}\right)\prod_{j=0,j\notin\{a,b\}}^{s}\exp\left(\frac{-j}{h_i}\right) - \cdots\right|\\ &\le \left|\sum_{0\le a\le s} O\left(\frac{a^2}{h_i^2}\right)\right| + \left|\sum_{0\le a<b\le s} O\left(\frac{a^2b^2}{h_i^4}\right)\right| + \left|\sum_{0\le a<b<c\le s} O\left(\frac{a^2b^2c^2}{h_i^6}\right)\right| + \cdots\\ &\le s\,O\left(\frac{s^2}{h_i^2}\right) + s^2\,O\left(\frac{s^4}{h_i^4}\right) + s^3\,O\left(\frac{s^6}{h_i^6}\right) + \cdots\\ &\le O\left(\frac{1}{h_i^{1/5}}\right) + O\left(\frac{1}{h_i^{2/5}}\right) + O\left(\frac{1}{h_i^{3/5}}\right) + \cdots\end{aligned}$$

Die Terme des ausmultiplizierten Produkts werden in der zweiten Zeile nach der Anzahl an $-O(j^2/h_i^2)$-Faktoren gruppiert. Von der zweiten zur dritten Zeile wird die Dreiecksungleichung verwendet und dass die Produke von $\exp(-j/h_i)$ kleiner gleich 1 sind. Anschließend ist alles positiv und die Beträge können weggelassen werden. Die letzte Zeile ist Teil einer geometrischen Reihe und geht daher für $h_i \to \infty$ gegen 0. (Eigentlich lassen wir $p \to \infty$ gehen, aber da $h_i = p/(\gcd(p-1, 2k_i) - 1)$ gilt $h_i = \Theta(p)$ und damit $h_i \to \infty \iff p \to \infty$.) Im Fall $s < h_i^{3/5}$ können wir also

$$\mathbb{P}(s_i > s) \sim \prod_{j=0}^{s}\exp\left(\frac{-j}{h_i}\right) = \exp\sum_{j=0}^{s}\frac{-j}{h_i} = \exp\frac{-s(s+1)}{2h_i} \sim \exp\frac{-s^2}{2h_i}$$

schreiben. Letzte Annäherung gilt für große s, für ausreichend große p sind aber fast alle s im Erwartungswert von $t_{\min}$ groß.

Nun wird $s > h_i^{3/5}$ angenommen und gezeigt, dass dann der Summand in $\mathbb{E}(t_{\min})$ für $p \to \infty$

gegen 0 geht. Hier gilt

$$\mathbb{P}(s_i > s) = \prod_{j=0}^{s} \exp\left(\frac{-j}{h_i}\right) \leq \prod_{j=0}^{\lfloor h_i^{3/5} \rfloor} \exp\left(\frac{-j}{h_i}\right) \sim \exp\frac{-\lfloor h_i^{3/5} \rfloor^2}{2h_i} \leq \exp\frac{-h_i^{1/5}}{2}$$

Die anderen Wahrscheinlichkeiten in dem entsprechenden Summanden sind alle durch 1 begrenzt und t ist $O(p)$. Da aber $h_i = \Theta(p)$ und $\lim_{p\to\infty} O(p)e^{-\Theta(p)^{1/5}} = 0$, geht der Summand gegen 0. Insgesamt bedeutet das, dass man durch Weglassen der $O(j^2/h_i^2)$-Terme entweder eine asymptotisch genaue Annäherung erhält, oder der Term, in dem man die Annäherung verwendet, sowieso gegen 0 geht. Als Zwischenergebnis erhalten wir

$$\begin{aligned}
\mathbb{E}_{t_{\min}} &\sim \sum_{t=0}^{t_{\max}} t \left(\prod_{i=1}^{M} \prod_{j=0}^{(t-1)/\lg 2k_i} \exp\left(\frac{-j}{h_i}\right) \right) \mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i) \\
&= \sum_{t=0}^{t_{\max}} t \exp\left(\sum_{i=1}^{M} \sum_{j=0}^{(t-1)/\lg 2k_i} \frac{-j}{h_i} \right) \mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i) \\
&\sim \sum_{t=0}^{t_{\max}} t \exp\left(\sum_{i=1}^{M} \frac{-t^2}{2h_i \lg^2 2k_i} \right) \mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i) \qquad (2)
\end{aligned}$$

$\mathbb{P}_{t_i>t-1\,\forall i}(t_i = t$ *für mindestens ein* $i)$. Die Wahrscheinlichkeit, dass bei Maschine i eine Kollision nach genau t Zeiteinheiten auftritt ist

$$\mathbb{P}_{t_i>t-1}(t_i = t) \sim \begin{cases} t/h_i \lg 2k_i & t = \lceil m \lg 2k_i \rceil \text{ für ein } m \in \mathbb{N} \\ 0 & t \neq \lceil m \lg 2k_i \rceil \text{ für jedes } m \in \mathbb{N} \end{cases}$$

Der erste Fall tritt ein, wenn die Zeiteinheit t das Ende einer Iteration von Maschine i enthält. Da bei Zeitpunkt t bereits $t/\lg 2k_i$ Schritte durchgeführt wurden, trifft Maschine i im nächsten Schritt mit Wahrscheinlichkeit $t/h_i \lg 2k_i$ auf einen bereits besuchten Knoten. Im zweiten Fall befindet sich Maschine i bei Zeiteinheit t mitten in einer Iteration, es kann also keine Kollision auftreten. Zur Bestimmung von $\mathbb{E}(t_{\min})$ kann das durch

$$\mathbb{P}_{t_i>t-1}(t_i = t) \sim \frac{t}{h_i \lg^2 k_i}$$

angenähert werden. Die Wahrscheinlichkeit einer Kollision an einem Zeitpunkt wird durch den zusätzlichen Faktor $1/\lg 2k_i$ auf die umliegenden Zeitpunkte „verteilt“. Pro Iteration von Maschine i gibt es statt einem Zeitpunkt mit Kollisionswahrscheinlichkeit $t/h_i \lg 2k_i$ nun $\lg 2k_i$ Zeitpunkte mit Kollisionswahrscheinlichkeit $t/h_i \lg^2 2k_i$. Dass das eine asymptotische Näherung ist, lässt sich damit begründen, dass der übrige Teil von (2) stetig in t ist und nicht schnell oszilliert oder Ähnliches. Ob die erste Kollision bei Maschine i also bei Zeitpunkt t oder $t + x$ für $|x| < \lg 2k_i$ auftritt macht daher asymptotisch keinen Unterschied. Die Wahrscheinlichkeit einer Kollision bei mindestens einer Maschine wird durch die Summe der Wahrscheinlichkeiten $\mathbb{P}_{t_i>t-1}(t_i = t)$ angenähert. Denn für alle relevanten t (d.h. $t \leq h_i^{3/5}\,\forall i$) ist die Wahrscheinlichkeit, dass zwei oder mehr Maschinen gleichzeitig bei Zeit t kollidieren für $p \to \infty$ verschwindend

gering. Insgesamt gilt also

$$\mathbb{P}_{t_i>t-1\,\forall i}(t_i = t \text{ für mindestens ein } i) \sim \sum_{i=1}^{M} \frac{t}{h_i \lg^2 2k_i} = t \sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \tag{3}$$

Für $\mathbb{E}(t_{\min})$ gilt mit (2) und (3)

$$\mathbb{E}(t_{\min}) \sim \left(\sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right) \sum_{t=0}^{t_{\max}} t^2 \exp\left(\frac{-t^2}{2} \sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right)$$

Als letzter Schritt wird die Summe über t durch ein Integral angenähert. Die intuitive Erklärung dafür, dass das am asymptotischen Wert nichts ändert, ist ähnlich wie eben. (2) und (3) sind stetig in t und schwanken nicht stark bei kleinen Veränderungen von t. Die Werte an einzelnen Punkten, wie sie in der Summe vorkommen, sind also nahezu gleich den Werten um diese Punkte herum, die zusätzlich im Integral vorkommen. Außerdem kann nach einem ähnlichen Argument wie dafür, dass Terme mit $t/\lg 2k_i > h_i^{3/5}$ asymptotisch irrelevant sind, die obere Integralgrenze bis ∞ geöffnet werden. Man erhält

$$\begin{aligned} L_{k_1,\ldots k_M}(p) = \mathbb{E}(t_{\min}) &\sim \left(\sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right) \int_0^\infty t^2 \exp\left(\frac{-t^2}{2} \sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right) dt \\ &= \left(\sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right) \sqrt{\pi/2} \left(\sum_{i=1}^{M} \frac{1}{h_i \lg^2 2k_i} \right)^{-3/2} \\ &= \sqrt{\pi p/2} \left(\sum_{i=1}^{M} \frac{\gcd(p-1, 2k_i) - 1}{\lg^2 2k_i} \right)^{-1/2} \end{aligned} \tag{4}$$

Zur Auswertung des Integrals wurde die Tabelle in Wikipedia: „Gaussian Integral“, 2023 verwendet. Man kann sich auf mehreren Wegen davon überzeugen, dass die Formel trotz der vielen Näherungen stimmt. Beispielsweise ist sie verträglich mit der bekannten Laufzeitabschätzung für Pollards Rho-Algorithmus im Fall $M = k_1 = 1$. Setzt man in (4) ein, erhält man $\sqrt{\pi p/2}$, wie Pollard, 1975, S. 332. Lässt man in obigem Integral einen Faktor t weg, erhält man statt des Erwartungswerts das Integral aller Wahrscheinlichkeiten. Dieses sollte natürlich 1 sein, und das ist es auch.

4 Der Fall zweier abhängiger Maschinen

Um $L_{k_1,\ldots,k_M}(p)$ zu bestimmen, wenn $k_i = k_j$ für $i \neq j$ gilt, muss die Abhängigkeit der Rho-Längen der i-ten und j-ten Maschinen berücksichtigt werden. Denn setzt man beispielsweise $M = 2$ und $k_1 = k_2 = 1$ in (4) ein, erhält man eine erwartete Laufzeit von $\sqrt{\pi p/4}$. In diesem Abschnitt wird allerdings gezeigt, dass unter Berücksichtigung der Abhängigkeit $25/32\sqrt{\pi p/2}$ Schritte benötigt werden. Der Fall abhängiger Maschinen hat sich als weitaus schwieriger herausgestellt, weil keine allgemeine Formel gefunden wurde. Jedoch konnte der Fall $M = 2$ gelöst werden. In diesem Fall lässt sich das Problem folgendermaßen angehen. Sei $k = k_1 = k_2,\ h = p/(\gcd(p-1, 2k) - 1)$ und $f : x \mapsto x^{2k} + 1$. Da die zwei Maschinen beide f verwenden, bewegen sie sich beide im funktionalen Graphen $\gamma(f)$. Nach der Random Mapping

Assumption ist die Verteilung der Rho-Längen von f für $p \to \infty$ asymptotisch zur Verteilung der Rho-Längen eines zufälligen Elements aus $A(h)$. Das Problem reduziert sich daher auf folgende Frage: *Gegeben sei ein zufälliges Element g aus $A(n)$ und zwei zufällige Elemente $a, b \in \mathbb{Z}/n\mathbb{Z}$ für $n \in \mathbb{N}$. Was ist der Erwartungswert von* $\min\{\nu(g,a), \nu(g,b)\}$*?* Die Frage wird von folgendem Satz beantwortet, dessen Beweis das Ziel der nächsten zwei Abschnitte ist.

Satz 1. *Sei $A(n)$ die Menge der Abbildungen $\mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$. Wir bezeichnen mit*

$$\tau_n = \frac{1}{n^{n+2}} \sum_{g \in A(n)} \sum_{a \in \mathbb{Z}/n\mathbb{Z}} \sum_{b \in \mathbb{Z}/n\mathbb{Z}} \min\{\nu(g,a), \nu(g,b)\}$$

die erwartete minimale Rho-Länge zweier zufälliger Knoten in einem funktionalen Graphen von Größe n. Es gilt

$$\tau_n \sim \frac{25}{32}\sqrt{\pi n/2}$$

Der Vorfaktor in der Definition von τ_n ist $1/n^{n+2}$, da es n^n Abbildungen $\mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ gibt und für jede von diesen n^2 Paare an Anfangswerten.

4.1 Theoretischer Hintergrund: Erzeugende Funktionen

Der Ansatz zum Beweis von Satz 1 ist, die Summe in der Definition von τ_n mithilfe einer erzeugenden Funktion $\Psi(x,w)$ zu bestimmen. $\Psi(x,w)$ zählt funktionale Graphen mit einem Paar an Knoten. Die Variable x markiert die Größe des Graphen und die Variable w die minimale Rho-Länge der zwei Anfangsknoten. Da funktionale Graphen beschriftet sind, werden stets erzeugende Funktionen von exponentiellem Typ (EF) verwendet. Also gilt

$$\Psi(x,w) = \sum_{n=0}^{\infty} \frac{x^n}{n!} \sum_{g \in A(n)} \sum_{a \in \mathbb{Z}/n\mathbb{Z}} \sum_{b \in \mathbb{Z}/n\mathbb{Z}} w^{\min\{\nu(g,a),\nu(g,b)\}} \tag{5}$$

Aus den Koeffizienten der Reihe von Ψ kann dann wie folgt τ_n bestimmt werden.

$$\tau_n = \frac{n!}{n^{n+2}}[x^n]\left(\frac{\partial}{\partial w}\Psi(x,w)\right)\Bigg|_{w=1}$$

wobei $[x^n]$ den n-ten Koeffizienten in der Reihenentwicklung des nachstehenden Terms bezeichnet. Um im Folgenden erzeugende Funktionen funktionaler Graphen zu konstruieren, werden einige Komponenten benötigt. Eine intuitive Erklärung ihrer Struktur sowie ein Beweis der Korrektheit findet man in Flajolet und Sedgewick, 2009, S. 129, 148. Von x wird ein einzelner Knoten repräsentiert. Die erzeugende Funktion eines Pfads ist $1/(1-x)$. Ein Baum kann rekursiv als ein Wurzelknoten zusammen mit einer Menge an Bäumen definiert werden. Die erzeugende Funktion eines Baumes $T(x)$ ist also implizit durch $T(x) = x \exp T(x)$ gegeben. Ein funktionaler Graph wird durch $F(x) = 1/(1-T(x))$ beschrieben.

4.2 Bestimmung der mittleren minimalen Rho-Länge

Beweis von Satz 1. Im Folgenden ist g immer die betrachtete Funktion und a und b die zwei ausgezeichneten Knoten. Der Plan ist, zunächst einen geschlossenen Ausdruck für $\Psi(x,w)$ zu

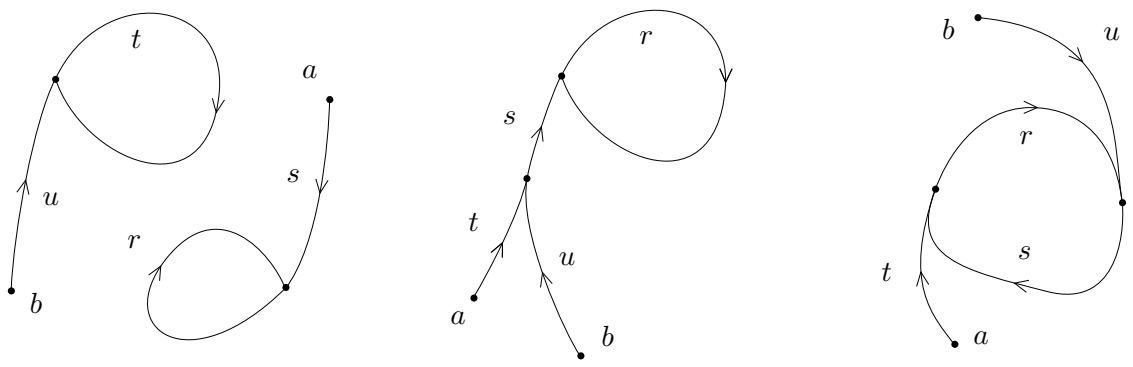

Abbildung 1: Die drei Fälle für die Bestimmung von $\Psi(x, w)$. Die Linien stellen keine einzelnen Kanten dar, sondern eine Folge an Kanten. Beispielsweise steht r in (α) für den Zyklus in der Zusammenhangskomponente von a.

bestimmen und dann die Ableitung nach w zu bilden. Dafür werden drei Fälle unterschieden, die in Abbildung 1 dargestellt sind. In jedem der Fälle wird zunächst eine EF für funktionale Graphen bestimmt, die nur die für a und b relevanten Teile enthalten. Der „relevante Teil“ besteht aus all den Knoten, die beim Ablaufen des Graphen besucht werden, also dem Zyklus und dem Pfad zum Zyklus. Der Rest des funktionalen Graphen wird später ergänzt. Grundsätzlich werden die Graphen so konstruiert, dass $\nu(g, a) \leq \nu(g, b)$ gilt. Erhält durch Vertauschen von a und b einen Fall, der noch nicht an anderer Stelle gezählt wurde, wird mit einem Faktor 2 multipliziert.

Fall 1. a und b liegen in unterschiedlichen Zusammenhangskomponenten. Dieser Fall wird erneut in die Fälle $\lambda(g, b) \leq \nu(g, a)$ und $\lambda(g, b) > \nu(g, a)$ unterteilt. Die erzeugende Funktion für den ersten Fall ist

$$\alpha_1(x, w) = \frac{x^2 w(1 + x^2 w)}{(1 - x^2 w)^3} \cdot \left(1 + \frac{2x}{1 - x}\right) = \frac{x^2 w(1 + x)(1 + x^2 w)}{(1 - x^2 w)^3 (1 - x)}$$

In diesem Fall ist es möglich, zuerst zwei ρ-Graphen mit gleicher Größe zu erzeugen und anschließend den Pfad von b zu seinem Zyklus zu verlängern. Ein ρ-Graph ist eine Zusammenhangskomponente in Abbildung 1 (α), d.h. ein Zyklus mit einem Pfad anhängend. Es gibt genau $n! \cdot n$ ρ-Graphen mit n Knoten, da es für jede Permutation der Knoten n Möglichkeiten für die Größe des Zyklus gibt. Folglich gibt es für gerade n genau $n! \cdot n^2/2$ Paare an ρ-Graphen, die beide $n/2$ Knoten besitzen. Die erzeugende Funktion von Paaren an ρ-Graphen gleicher Größe ist also

$$\sum_{n=0}^{\infty} x^n \, \frac{n^2}{4} \, \frac{1 + (-1)^n}{2} = \frac{x^2(1 + x^2)}{(1 - x^2)^3}$$

Durch einen der beiden ρ-Graphen wird die Zusammenhangskomponente von a erzeugt, durch den anderen t und der Anfang von u in Abbildung 1 (α). Da x die Zahl an Knoten markiert, wird von w die halbe Zahl an Knoten markiert, wenn x durch $x\sqrt{w}$ ersetzt wird. Dadurch erhält man die EF $x^2 w(1 + x^2 w)/(1 - x^2 w)^3$, in der w die Rho-Länge von a markiert. Das erklärt den ersten Faktor in $\alpha_1(x, w)$. Nun gibt es zwei Möglichkeiten: Wird u nicht verlängert, gilt $\nu(g, a) = \nu(g, b)$. Wird hingegen ein nichtleerer Pfad angehängt, dessen erzeugende Funktion

$x/(1-x)$ ist, ergeben sich zwei Möglichkeiten durch Vertauschen von a und b.

Im Fall $\lambda(g,b) > \nu(g,a)$ ist die erzeugende Funktion

$$\alpha_2(x,w) = 2\,\frac{x^2w}{(1-x^2w)^2}\,\frac{x}{1-x}\,\frac{1}{1-x} = \frac{2x^3w}{(1-x^2w)^2(1-x)^2}$$

Der Faktor x^2w repräsentiert die zwei Knoten, an denen a und b jeweils ihren Zyklus betreten, und der Knoten von a ist mit w markiert. Mit $1/(1-x^2w)^2$ erhält man vier Pfade r, y, s, z, sodass die Länge von r gleich der Länge von y und die Länge von s gleich der Länge von z ist. Die Summe der Längen von r und s wird von w markiert. r und s werden wie in Abbildung 1 (α) für die Zusammenhangskomponente von a verwendet. Der Zyklus t besteht aus y, z und einem Pfad von Länge ≥ 1, sodass $\lambda(g,b) > \nu(g,a)$ gilt. u wird von dem Term $1/(1-x)$ gebildet.

Fall 2. In diesem Fall liegen a und b im gleichen Baum und ihr kleinster gemeinsamer Vorfahre ist nicht die Wurzel. Die erzeugende Funktion lautet

$$\beta(x,w) = xw\,\frac{xw}{1-xw}\,\frac{1}{1-xw}\,\frac{1}{1-x^2w}\left(1+\frac{2x}{1-x}\right) = \frac{x^2w^2(1+x)}{(1-xw)^2(1-x^2w)(1-x)}$$

Der Zyklusknoten, an dem der Baum von a und b anhängt, wird durch xw repräsentiert. Der Pfad s in Abbildung 1 (β) wird von $xw/(1-xw)$ erzeugt. Er muss mindestens Länge 1 haben, da der kleinste gemeinsame Vorfahre von a und b sonst die Wurzel wäre. Der Faktor $1/(1-xw)$ steht für den Zyklus r. Mit $1/(1-x^2w)$ werden zwei gleich lange Pfade erzeugt, deren einfache Länge durch w markiert wird. Einer von ihnen wird für t verwendet und der andere ist ein Teil von u. Nun gibt es wie in Fall 1 wieder die Option, u echt zu verlängern und so einen Faktor 2 für die mögliche Vertauschung von a und b zu erhalten, oder ihn zu lassen, wobei es dann wegen der Symmetrie um a und b nur eine Möglichkeit gibt.

Fall 3. Zuletzt bleibt der Fall, wenn a und b in verschiedenen Bäumen liegen oder die Wurzel ihr kleinster gemeinsamer Vorfahre ist. Eine andere Sichtweise ist, dass sich die Pfade von a und b bei wiederholtem Anwenden von g erstmals bei einem Zyklusknoten treffen. Die erzeugende Funktion ist hier

$$\gamma(x,w) = xw\,\frac{1}{(1-xw)^2}\,\frac{1}{1-x^2w}\left(1+\frac{2x}{1-x}\right) = \frac{xw(1+x)}{(1-xw)^2(1-x^2w)(1-x)}$$

Der Knoten, an dem a den Zyklus betritt, ist xw. Der Term $1/(1-xw)^2$ erzeugt zwei Pfade, deren Länge mit w markiert wird. Die zwei Pfade sind die zwei Teile r und s des Zyklus in Abbildung 1 (γ). Ähnlich wie in Fall 2 ist der Term $1/(1-x^2w)$ ein Paar an gleich langen Pfaden, deren Länge von w markiert wird. Einer der Pfade ist t und der andere ein Teil von u. Auch hier kann man die Länge von u unverändert lassen, in diesem Fall gibt es eine Möglichkeit, oder einen Pfad von Länge ≥ 1 hinzufügen, sodass es zwei Möglichkeiten durch Vertauschung von a und b gibt.

Ein funktionaler Graph besteht natürlich nicht nur aus einem Zyklus und den Pfaden von a und b zum Zyklus. Von jedem Knoten kann ein Baum ausgehen und es kann noch weitere Zusammenhangskomponenten geben. Indem x durch $T(x)$ ersetzt wird, kann von jedem Knoten ein Baum ausgehen. Weitere Zusammenhangskomponenten bilden einen funktionalen Graphen, sie können also durch Multiplikation der EF mit $F(x)$ hinzugefügt werden. Weil die drei Fälle

disjunkt sind und zusammen alle Möglichkeiten abdecken, gilt

$$\Psi(x,w) = (\alpha_1(T(x),w) + \alpha_2(T(x),w) + \beta(T(x),w) + \gamma(T(x),w))\,F(x)$$

Damit erhalten wir

$$\psi(x) = \left(\frac{\partial}{\partial w}\Psi(x,w)\right)\Bigg|_{w=1} = \frac{T(x)(1+2T(x)+2(T(x))^2)(1+5T(x)+3(T(x))^2+(T(x))^3)}{(1-T(x))^6(1+T(x))^3}$$

Nun soll die analytische Methode von Flajolet und Odlyzko, 1990 verwendet werden, um eine asymptotische Abschätzung für die Koeffizienten der Taylorreihe von $\psi(x)$ zu erhalten. Nach Flajolet und Odlyzko, 1990, S. 334, Proposition 1 ist die betragsmäßig kleinste Singularität von $T(x)$ in $\mathbb{C}$ bei $x = e^{-1}$ und es gilt

$$T(x) \sim 1 - \sqrt{2}\,\sqrt{1-ex}$$

für $x \to e^{-1}$. $\psi(x)$ hat keine betragsmäßig kleineren Singularitäten, denn wenn $1 - T(x) = 0$, rechnet man leicht nach, dass $x = e^{-1}$ gilt, und wenn $1 + T(x) = 0$, gilt $x = -e$. Es wird nun Theorem 1 aus Flajolet und Odlyzko, 1990, S. 333 mit $s = e^{-1}$ verwendet. Wenn $x \to e^{-1}$, gilt

$$\begin{aligned}
\psi(x) &\sim \frac{T(e^{-1})(1+2T(e^{-1})+2(T(e^{-1}))^2)(1+5T(e^{-1})+3(T(e^{-1}))^2+(T(e^{-1}))^3)}{(1-(1-\sqrt{2}\cdot\sqrt{1-ex}))^6(1+T(e^{-1}))^3} \\
&= \frac{(1+2+2)(1+5+3+1)}{(\sqrt{2})^6(\sqrt{1-ex})^6\cdot 2^3} \\
&= \frac{50}{64(1-ex)^3}
\end{aligned}$$

Folglich gilt mit der Notation in Flajolet und Odlyzko, 1990 $\sigma(x) = x^3$ und $\alpha = 3$. Aus Theorem 1 folgt

$$[x^n]\psi(x) \sim \frac{50}{64}\,(e^{-1})^{-n}\,\frac{n^3}{n\Gamma(3)} = \frac{25}{32}\,\frac{e^n n^2}{2}$$

und mit Stirlings Näherung $n! \sim \sqrt{2\pi n}(n/e)^n$

$$\tau_n \sim \frac{n!}{n^{n+2}}\,\frac{25}{32}\,\frac{e^n n^2}{2} = n!\left(\frac{e}{n}\right)^n\frac{25}{64} \sim \sqrt{2\pi n}\,\frac{25}{64} = \frac{25}{32}\sqrt{\pi n/2}$$

□

Aus Satz 1 folgt sofort, dass $L_{1,1}(p) \sim 25/32\sqrt{\pi p/2}$. Daraus lässt sich auch auf $L_{k,k}(p)$ schließen. Für $1 \leq k \in \mathbb{N}$ wird nach RMA p durch $p/(\gcd(p-1,2k)-1)$ ersetzt. Die Laufzeit pro Iteration steigt bei jeder Maschine aber um einen Faktor $\lg 2k$, sodass sie auch insgesamt um einen Faktor $\lg 2k$ steigt. Es gilt also

$$L_{k,k}(p) \sim 25/32\sqrt{\pi p/2}\;\frac{\lg 2k}{\sqrt{\gcd(p-1,2k)-1}} \tag{6}$$

Man bemerke, dass Satz 1 unabhängig von der Anwendung auf Pollards Rho-Algorithmus formuliert wurde und nicht auf der Random Mapping Assumption basiert.

5 Bestimmung optimaler Exponenten für die Rho-Methode

In diesem Abschnitt wird die Frage behandelt, wie der Paramter k bei M Maschinen bestmöglich gewählt werden kann. Mit Formel (4) und Satz 1 konnten Ergebnisse in den Fällen $M = 1$ und $M = 2$ erzielt werden. Die grundlegende Strategie ist, den Erwartungswert von $L_{k_1,\ldots,k_M}(p)$ über alle Möglichkeiten von $\gcd(p-1, 2k_i)$ für alle $1 \leq i \leq M$ zu bilden und so einen Wert für die erwartete Laufzeit in Abhängigkeit der k_i zu erhalten. Da $p-1$ gerade ist, gilt $\gcd(p-1, 2k_i) = 2\gcd((p-1)/2, k_i)$. Weitere Kongruenzen von $p-1$ sind im Allgemeinen nicht bekannt, weshalb angenommen wird, dass jeder Rest $(p-1)/2$ modulo k_i gleich wahrscheinlich ist.

5.1 Der Fall einer Maschine

Satz 2. *Sei $L_k(p)$ wie in (1) definiert. Unter RMA gilt $\mathbb{E}(L_1(p)) < \mathbb{E}(L_k(p))$, wobei $1 < k \in \mathbb{N}$ und der Erwartungswert über alle möglichen $\gcd((p-1)/2, k)$ genommen wird.*

Beweis. Durch Einsetzen von $M = 1$ in (4) erhalten wir

$$L_k(p) \sim \sqrt{\pi p/2}\,\frac{\lg 2k}{\sqrt{\gcd(p-1, 2k) - 1}}$$

Die erwartete Laufzeit im Fall $k = 1$ ist folglich $\sqrt{\pi p/2}$. Es wird also gezeigt, dass $\mathbb{E}(L_k(p)) > \sqrt{\pi p/2}$ für $k > 1$. Mit φ wird die eulersche Phifunktion bezeichnet. Dann gilt

$$\mathbb{E}(L_k(p)) \sim \sqrt{\pi p/2}\,\lg(2k) \sum_{d|k} \frac{\mathbb{P}(\gcd((p-1)/2, k) = d)}{\sqrt{2d-1}} \geq \sqrt{\pi p/2}\,\lg(2k)\,\frac{\varphi(k)}{k}$$

Für die Ungleichung wurde statt der Summe über alle Teiler nur $d = 1$ betrachtet. Da es $\varphi(k)$ teilerfremde Zahlen kleiner k gibt, ist $\mathbb{P}(\gcd((p-1)/2, k) = 1) = \varphi(k)/k$. Nach Rosser und Schoenfeld, 1962, Theorem 15 gilt $\varphi(k)/k > 1/(e^\gamma \ln(\ln(k)) + 2.51/\ln(\ln(k)))$ für $k \geq 3$, wobei $\gamma \approx 0.5772$ die Euler-Mascheroni-Konstante ist. Für $k \geq e^e$ folgt daraus $\varphi(k)/k > 1/(e^\gamma \ln(\ln(k)) + 2.51)$. Indem nun gezeigt wird, dass $\lg(2k)/(e^\gamma \ln(\ln(k)) + 2.51) > 1$ für $k \geq 16$ wird der Satz im Fall $k \geq 16$ bewiesen. Sei $f(x) = \lg(2x)/(e^\gamma \ln(\ln(x)) + 2.51)$. Es gilt $f(16) \approx 1.1557$ und

$$f'(x) = \frac{\ln(x)(\ln(\ln(x)) + 1.51) - \ln 2}{x \ln(2) \ln(x)(\ln(\ln(x)) + 2.51)^2}$$

Für $x \geq 16$ ist der Nenner von f' positiv, denn $x > 0$ und $\ln x > 0$, und weil $\ln\ln x > 1$ für $x \geq 16$ ist der Term unter dem Quadrat positiv. Der Zähler ist ebenfalls positiv, da $\ln x \geq 1$ und $\ln\ln x \geq 1$, woraus $\ln(x)(\ln(\ln(x)) + 1.51) \geq 1 \cdot (1 + 1.51) = 2.51 > \ln 2$ folgt. Also ist f streng monoton steigend für $x \geq 16$, und da bereits $f(16) > 1$ gezeigt wurde, folgt $f(x) > 1$ für $x \geq 16$. Der Fall $k < 16$ wurde durch Ausrechnen von (4) für $2 \leq k \leq 15$ überprüft. □

5.2 Der Fall zweier Maschinen

Satz 3. *Sei $L_{k_1,k_2}(p)$ wie in Abschnitt (1) definiert. Man nehme RMA an und bilde den Erwartungswert $\mathbb{E}(L_{k_1,k_2})$ über alle möglichen $\gcd((p-1)/2, k_i)$ für $i = 1, 2$.*

1. *Wenn k_1, k_2 unterschiedliche Primzahlen sind, gilt $\mathbb{E}(L_{1,1}(p)) < \mathbb{E}(L_{k_1,k_2}(p))$.*

2. Wenn $1 < k \in \mathbb{N}$, gilt $\mathbb{E}(L_{1,1}(p)) < \mathbb{E}(L_{k,k}(p))$.

Beweis. Nach Satz 1 gilt $L_{1,1}(p) \sim 25/32\sqrt{\pi p/2}$, es gilt also in beiden Teilen zu zeigen, dass der Erwartungswert größer ist. Zunächst wird Teil 1 bewiesen. Da $k_1 \neq k_2$, sind die zwei Maschinen unabhängig und (4) kann verwendet werden.

$$L_{k_1,k_2}(p) \sim \sqrt{\pi p/2}\left(\frac{\gcd(p-1,2k_1)-1}{\lg^2 2k_1} + \frac{\gcd(p-1,2k_2)-1}{\lg^2 2k_2}\right)^{-1/2}$$

Durch Bilden des Erwartungswerts über alle möglichen $\gcd((p-1)/2, k_i)$ für $i = 1, 2$ erhält man

$$\begin{aligned}
\mathbb{E}(L_{k_1,k_2}) &\sim \sqrt{\pi p/2} \sum_{d_1|k_1} \mathbb{P}(\gcd((p-1)/2, k_1) = d_1) \\
&\qquad \sum_{d_2|k_2} \mathbb{P}(\gcd((p-1)/2, k_2) = d_2)\left(\frac{2d_1-1}{\lg^2 2k_1} + \frac{2d_2-1}{\lg^2 2k_2}\right)^{-1/2} \\
&\geq \sqrt{\pi p/2}\ \frac{\varphi(k_1)\varphi(k_2)}{k_1 k_2}\left(\frac{1}{\lg^2 2k_1} + \frac{1}{\lg^2 2k_2}\right)^{-1/2} \\
&= \sqrt{\pi p/2}\ \frac{(k_1-1)(k_2-1)}{k_1 k_2}\sqrt{\frac{\lg^2(2k_1)\lg^2(2k_2)}{\lg^2(2k_1)+\lg^2(2k_2)}} \qquad (7)
\end{aligned}$$

Wie bei $M = 1$ wurden die Summen über alle Teiler von k_1, k_2 durch den Wert für $d_1 = d_2 = 1$ nach unten begrenzt. Die Terme $(k_i - 1)/k_i$ sind streng monoton steigend in k_i. Ebenso ist das Argument der nachfolgenden Wurzel in (7) streng monoton steigend in jedem der k_i. Um das zu zeigen, sei $x = \lg^2 2k_1, y = \lg^2 2k_2$ und $x' > x$. Dann gilt

$$\frac{x'y}{x'+y} = \frac{x'y}{x'+y}\,\frac{x+y}{xy}\,\frac{xy}{x+y} = \frac{xx'y + x'y^2}{xx'y + xy^2}\,\frac{xy}{x+y} > \frac{xy}{x+y}$$

da $x, x', y > 0$ und $x < x'$. Der Term ist symmetrisch in x und y, womit er auch streng monoton steigend in y ist. Da die Wurzelfunktion streng monoton steigt und die Verkettung streng monoton steigender Funktionen streng monoton steigt, folgt, dass die gesamte Wurzel auf der rechten Seite von (7) streng monoton steigt. Weil nun jeder einzelne Faktor in (7) streng monoton steigt und positiv ist, ist ganz (7) streng monoton steigend in den k_i. Indem man $k_1 = 3,\ k_2 = 5$ in (7) einsetzt, sieht man, dass $E(L_{3,5}(p)) \geq 1.0881\sqrt{\pi p/2} > 25/32\sqrt{\pi p/2}$. Weil (7) symmetrisch in k_1, k_2 ist, kann $k_1 < k_2$ angenommen werden. Dann folgt aus der Monotonie von (7), dass $E(L_{k_1,k_2}(p)) > \mathbb{E}(L_{1,1}(p))$, wenn $k_1 \geq 3$. Es bleibt also lediglich der Fall $k_1 = 2$. Durch Einsetzen von $k_1 = 2,\ k_2 = 11$ in (7) gilt $\mathbb{E}(L_{2,11}(p)) \geq 0.8294\sqrt{\pi p/2} > 25/32\sqrt{\pi p/2}$. Aus der Monotonie von (7) folgt $\mathbb{E}(L_{k_1,k_2}(p)) > \mathbb{E}(L_{1,1}(p))$ für $k_1 = 2$ und $k_2 \geq 11$. Die übrigen Fälle $k_1 = 2$ und $k_2 = 3, 5, 7$ wurden nachgerechnet.

Nun zum Beweis von Teil 2. Aus (6) folgt $L_{k,k}(p) = 25/32 \cdot L_k(p)$ für $1 \leq k \in \mathbb{N}$. Damit folgt Teil 2 des Satzes aus Satz 3. □

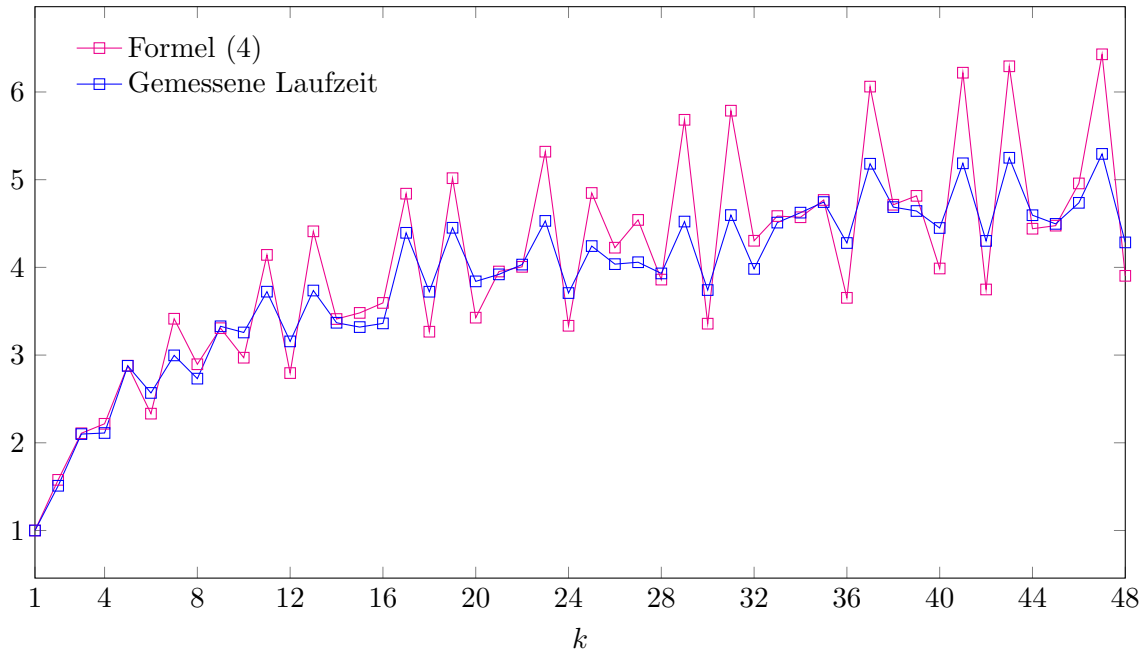

Abbildung 2: Die durchschnittliche gemessene Laufzeit des Rho-Algorithmus für $M = 1$ und Werte von Formel (4) für $1 \leq k \leq 48$.

$k_1 \backslash k_2$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1.00	1.03	1.08	1.08	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09
2		1.55	1.55	1.57	1.63	1.62	1.63	1.63	1.65	1.65	1.65	1.65	1.65	1.66
3			2.18	2.05	2.16	2.15	2.18	2.15	2.24	2.21	2.24	2.22	2.24	2.22
4				2.18	2.17	2.15	2.24	2.19	2.23	2.24	2.25	2.23	2.27	2.25
5					2.94	2.64	2.76	2.69	2.81	2.84	2.89	2.77	2.90	2.83
6						2.72	2.66	2.65	2.71	2.69	2.72	2.71	2.72	2.70
7							3.03	2.75	2.89	2.87	2.97	2.83	2.97	2.92
8								2.84	2.76	2.78	2.82	2.76	2.82	2.80
9									3.46	3.24	3.33	3.25	3.33	3.26
10										3.42	3.31	3.22	3.31	3.28
11											3.75	3.26	3.50	3.38
12												3.32	3.26	3.24
13													3.76	3.38
14														3.51

$k_1 \backslash k_2$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1.00	1.06	1.12	1.12	1.17	1.13	1.19	1.16	1.18	1.17	1.22	1.15	1.22	1.19
2		1.58	1.51	1.53	1.66	1.55	1.73	1.64	1.70	1.66	1.81	1.61	1.83	1.71
3			2.11	1.78	2.01	1.81	2.14	1.97	2.07	2.00	2.27	1.92	2.31	2.10
4				2.22	2.05	1.85	2.19	2.02	2.12	2.04	2.34	1.96	2.38	2.15
5					2.88	2.10	2.61	2.35	2.51	2.38	2.83	2.27	2.90	2.55
6						2.33	2.25	2.06	2.18	2.09	2.41	2.00	2.46	2.21
7							3.41	2.56	2.76	2.60	3.16	2.46	3.26	2.81
8								2.90	2.47	2.34	2.79	2.23	2.86	2.51
9									3.31	2.51	3.04	2.38	3.13	2.70
10										2.97	2.83	2.26	2.91	2.55
11											4.14	2.68	3.67	3.09
12												2.79	2.74	2.42
13													4.41	3.18
14														3.41

Abbildung 3: Die durchschnittliche Laufzeit von Pollards Rho-Algorithmus (oben) und berechnete Werte für die erwartete Laufzeit (unten), für $M = 2$ und $1 \leq k_1 \leq k_2 \leq 14$.

6 Experimentelle Ergebnisse

Um zu demonstrieren, dass Formel (4), Satz 2 und Satz 3 das Laufzeitverhalten von Pollards Rho-Algorithmus gut beschreiben, wurden Laufzeitmessungen für $M = 1$ und $M = 2$ durchgeführt. Wie in Abschnitt 2.1 gesagt, wurde für die Berechnungen angenommen, dass n genau einen kleinsten Primfaktor p hat und die anderen Primfaktoren deutlich größer sind, sodass sie zur Analyse der Laufzeit ignoriert werden können. Es ist daher nur sinnvoll, die Berechnungen mit Messergebnissen für Zahlen dieser Art zu vergleichen. Als Testzahlen wurden 62-Bit Semiprimzahlen mit einem 21-Bit und einem 41-Bit Faktor verwendet. Eine Semiprimzahl ist das Produkt zweier Primzahlen. Für jede Parameterwahl wurden 2^{20} zufällige Semiprimzahlen gewählt und die durchschnittliche Laufzeit berechnet. Die berechneten und gemessenen Werte wurden jeweils normalisiert, sodass bei $k = 1$ bzw. $k_1 = k_2 = 1$ der Wert 1 ist. Bei den Messungen wurden Ausreißer entfernt, die für alle Parameterwerte jeweils weniger als 0.01% der Proben ausmachten.

Eine Maschine. Die durchschnittliche Laufzeit für $1 \leq k \leq 48$ und Werte von (4) zum Vergleich sind in Abbildung 2 dargestellt. Die Laufzeit wird gut von (4) angenähert, denn die Werte befinden sich in der gleichen Größenordnung und auch Charakteristika spezieller Zahlen, wie beispielsweise hohe Werte bei Primzahlen, werden von beiden reflektiert.

Zwei Maschinen. Die durchschnittliche Laufzeit und berechnete Werte für jedes Paar $1 \leq k_1 \leq k_2 \leq 14$ sind in Abbildung 3 dargestellt. Zur Berechnung der erwarteten Laufzeit wurde Formel (6) verwendet, wenn $k_1 = k_2$, und Formel (4), wenn $k_1 \neq k_2$. Auch hier wird das grundsätzliche Verhalten der Laufzeit gut durch (4) und (6) beschreiben. Die Aussage von Satz 3 wird bestätigt, und es gibt zumindest für $1 \leq k_1 \leq k_2 \leq 14$ kein Paar k_1, k_2 mit einer geringeren Laufzeit als $k_1 = k_2 = 1$. Da die Laufzeit für größere k_1, k_2 tendenziell zu steigen scheint, wird die Vermutung aufgestellt, dass $k_1 = k_2 = 1$ optimal ist. Auffällig ist, dass die Laufzeitunterschiede weniger ausgeprägt sind als von den Formeln vorhergesagt. Das liegt wahrscheinlich daran, dass neben der Berechnung von x^{2k} auch andere Berechnungen im Rho-Algorithmus durchgeführt werden, deren Dauer unabhängig von k ist (z.B. das Bilden des gcd). Diese werden aber in den Formeln nicht berücksichtigt.

7 Fazit

Zur Beantwortung der Frage nach der optimalen Wahl des Parameters k konnten in dieser Arbeit grundlegende Formeln und Methoden entwickelt werden. Es wurde der Fall betrachtet, dass die zu faktorisierende Zahl n nur einen Primfaktor kleinster Größenordnung hat, sodass der Einfluss anderer Primfaktoren vernachlässigt werden kann. In diesem Fall konnte unter üblichen Annahmen über Pollards Rho-Algorithmus eine Formel für die erwartete Laufzeit aufgestellt werden, wenn die k_i paarweise verschieden sind. Auch konnte die erwartete Rho-Länge im Fall zweier abhängiger Maschinen bestimmt werden. Die damit erzielten Ergebnisse für $M = 1$ und $M = 2$ werden von Laufzeitmessungen unterstützt. Für eine vollständige Beantwortung der Frage sind allerdings noch einige Schritte nötig. Beispielsweise ist es für $M \geq 3$ möglich, dass manche Maschinen abhängig sind und manche unabhängig (wenn z. B. $k_1 = k_2 \neq k_3$). Diese Situation kann mit den hier entwickelten Methoden noch nicht behandelt werden.

Literatur

Brent, R. P. (1980). An improved Monte Carlo factorization algorithm. *BIT Numerical Mathematics*, *20*(2), 176–184. https://doi.org/10.1007/BF01933190

Brent, R. P., & Pollard, J. M. (1981). Factorization of the eighth Fermat number. *Mathematics of Computation*, *36*, 627–630. https://doi.org/10.1090/S0025-5718-1981-0606520-5

Flajolet, P., & Odlyzko, A. M. (1990). Random Mapping Statistics. In J.-J. Quisquater & J. Vandewalle (Hrsg.), *Advances in Cryptology — EUROCRYPT '89* (S. 329–354). Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-46885-4_34

Flajolet, P., & Sedgewick, R. (2009). *Analytic Combinatorics*. Cambridge University Press. https://algo.inria.fr/flajolet/Publications/book.pdf

Gaussian Integral. (2023). https://en.wikipedia.org/wiki/Gaussian_integral

Knuth, D. E. (1998). *The Art of Computer Programming - Volume 2 (Seminumerical Algorithms)*. Addison-Wesley.

Pollard, J. M. (1975). A monte carlo method for factorization. *BIT Numerical Mathematics*, *15*(3), 331–334. https://doi.org/10.1007/BF01933667

Rosser, J. B., & Schoenfeld, L. (1962). Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, *6*(1), 64–94. https://doi.org/10.1215/ijm/1255631807