Doppelnebenklassenvertreter einer maximalen Untergruppe in Fi_{23}

Max Neunhöffer
Lehrstuhl D für Mathematik
RWTH Aachen

Computeralgebra-Tagung 2005

Überblick

Problemstellung

$G := \mathrm{Fi}_{23} = \langle a, b \rangle$ mit $|G| = 4\,089\,470\,473\,293\,004\,800$,

$N = \langle n_1, n_2, n_3 \rangle \leq G$ mit $|N| = 3\,265\,173\,504$,
wobei die n_i als Worte in a und b gegeben sind.

Bekannt: $G = Ng_1N \dot\cup Ng_2N \dot\cup \cdots \dot\cup Ng_{36}N$,
wobei $NgN = \{n \cdot g \cdot n' \mid n, n' \in N\}$ *ist*.

Problem: Finde $\{g_1, \ldots, g_{36}\}$ als Worte in a und b.

Anwendung:
Es sei $K \triangleleft N$ und $F := GF(2)$ und $2 \nmid |K|$.

Dann gilt für $e^2 = e := \frac{1}{|K|} \sum_{k \in K} k \in FG$:

$$eFGe = \langle eg_1e, \ldots, eg_{36}e, en_1e, en_2e, en_3e \rangle_{F-\mathrm{Alg.}}$$

(F. Noeske, 2005)

Doppelnebenklassen und Suborbits

$G = Ng_1N \dot\cup Ng_2N \dot\cup \cdots \dot\cup Ng_{36}N$

G operiert auf der Menge $G/N := \{gN \mid g \in G\}$

$G/N = N \cdot g_1N \dot\cup \cdots \dot\cup N \cdot g_{36}N$

Also: Doppelnebenklassen $\leftrightarrow$ Suborbits

Probleme:

- $|G/N| = 1\,252\,451\,200 \approx 1.25 \cdot 10^9$
- Permutationen für a, b, n_1, n_2, n_3 bräuchten je etwa 5 GB
- Nicht leicht zu bestimmen!
- G als Permutationsgruppe auf 31 671 Punkte gegeben
- Elemente g_i zu bestimmen wäre langwierig

Realisierung der Operation auf G/N

Lineare Darstellung von G auf $V := F^{1494\times 1}$ ($F = GF(2)$):

$$\text{Gruppenhomomorphismus} \quad D : G \to \mathrm{GL}_{1494}(F)$$

Finde Vektor $v_1 \in V$ mit $D(N) \cdot v_1 = \{v_1\}$.

$\Longrightarrow$ Bahn $D(G) \cdot v_1$ ist isomorph zu G/N als G-Menge.

Wir können dadurch:

- Punkte von G/N als Vektoren in V speichern und vergleichen
- Gruppenelemente (Worte in a, b) anwenden

Immer noch zu groß:

- Jeder Vektor braucht etwa 200 bytes ($\approx 1494/8$)
- Insgesamt rund 250 GB (Hauptspeicher!)
- Es dauert ungemütlich lange, alle Vektoren aufzuzählen!

Ein Trick

Es sei $U < N$ mit $|U| = 6561$.

Idee: Gehe U-Bahn-weise vor:

- N-Bahnen zerfallen in U-Bahnen
- Zähle U-Bahnen auf

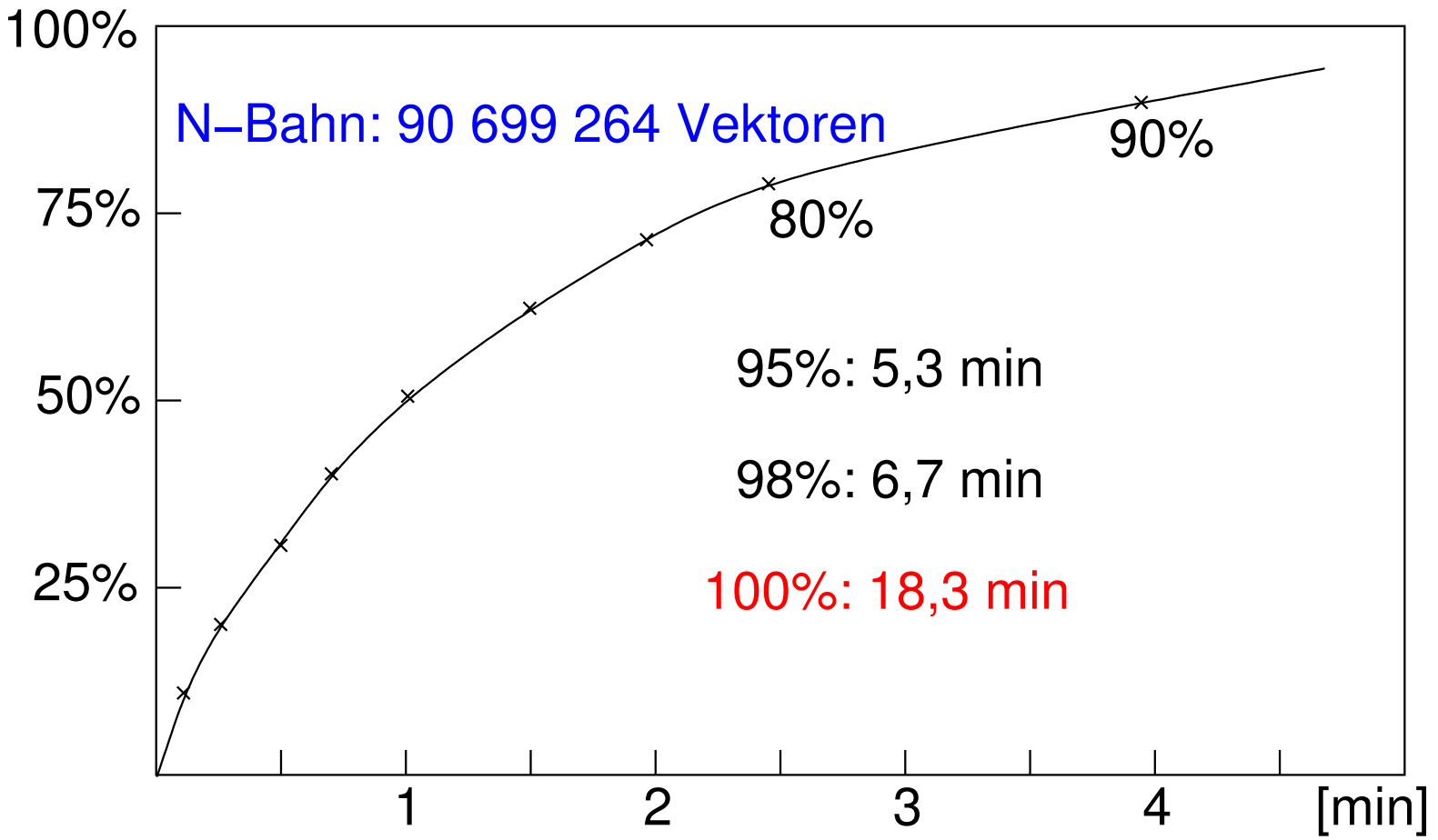
Dazu:

- Wähle in jeder U-Bahn B eine Teilmenge $\min(B) \subseteq B$ aus ("U-minimale" Vektoren) und zwar so, dass
- wir Algorithmus haben, der für jede U-Bahn B und zu jedem $v \in B$ ein $u(v) \in U$ liefert mit $D(u(v)) \cdot v \in \min(B)$.
- Speichere B durch Speichern von $\min(B)$

$\Longrightarrow$ Spare etwa Faktor 250 an Speicher und Zeit!

Immer noch langwierig

Verlauf einer N-Bahnaufzählung nach U-Bahnen:



Halbe Bahnen

Wir zählen nur die Hälfte jeder N-Bahn auf!

Angenommen, wir kennen $H \subseteq D(N) \cdot v$ mit $|H| > |D(N) \cdot v|/2$.

Frage: Liegt $w \in V$ in der Bahn $D(N) \cdot v$?

Antwort:
Wende zufällige Elemente $\{m_1, \ldots, m_{40}\}$ aus N auf w an und teste, ob $\{D(m_i) \cdot w \mid 1 \leq i \leq 40\} \cap H \neq \emptyset$ ist.

Wenn ja: w liegt mit Sicherheit in $D(N) \cdot v$

Wenn nein: w liegt wahrscheinlich nicht in $D(N) \cdot v$

$\Longrightarrow$ Reicht, um verschiedene N-Bahnen in $D(G) \cdot v_1$ zu finden.

Die ersten 35 sind jetzt machbar

N-Suborbitlängen in $D(G) \cdot v_1$:

1	10 077 696	20 155 392	3 888
22 674 816	90 699 264	5 038 848	78 732
34 012 224	10 077 696	3 359 232	19 683
272 097 792	68 024 448	5 038 848	186 624
90 699 264	136 048 896	7 558 272	62 208
90 699 264	30 233 088	3 779 136	124 416
272 097 792	30 233 088	3 779 136	15 552
10 077 696	944 784	1 679 616	15 552
944 784	30 233 088	1 679 616	768

Wende Bahnaufzählung für $D(G) \cdot v_1$ an, N-Bahn-weise.

Wir finden alle N-Bahnen, bis auf die mit 768 Vektoren.

Noch nicht bewiesen, dass diese 35 Bahnen verschieden sind!

Rasterfahndung

Wir suchen einen von 768 Vektoren v mit folgenden Eigenschaften:

- $v \in D(G) \cdot v_1$
- $S := Stab_N(v) < N$ hat Index 768 in N

Vorgehensweise:

- Rate $S < N$ mit $[N : S] = 768$ (nicht eindeutig!)
- Berechne Kandidaten $C := \{v \in V \mid D(S) \cdot v = \{v\}\}$
- Checke für alle $v \in C$, ob $D(a) \cdot v$ in einer der 35 (halb) bekannten N-Bahnen liegt
- Wenn ja, folgt $v \in D(G) \cdot v_1$ (bewiesen!).

  Ist v selbst nicht in einer der 35 Bahnen, so sind wir fertig.
- $\Longrightarrow$ liefert tatsächlich Vektor v_{36}

Der letzte Vertreter

Brauchen noch ein Wort g_{36} in a und b, das v_1 auf v_{36} abbildet!

Gehe so vor:

- Zähle mit Breitensuche durch Anwenden von a und b Vektoren in $D(G) \cdot v_1$ auf, bis der Speicher voll ist
- Suche rückwärts einen bekannten Vektor von v_{36} aus mit Tiefensuche (wende a^{-1} und b^{-1} an)
- Setze Vorwärts- und Rückwärtswort zusammen
- $\Longrightarrow$ findet Wort in wenigen Minuten
- Verbesserungsmöglichkeit: “nach U-Bahnen” (hier nicht nötig)

Verifikation

Für Kandidaten für $g_1, \ldots, g_{36}$:

Zähle alle N-Suborbits $D(N) \cdot D(g_i) \cdot v_1$ komplett auf

Braucht etwa 3 h auf einem Rechner mit 4 GB Hauptspeicher

$\Longrightarrow$ Bewiesen:
Alle N-Bahnen liegen in $D(G) \cdot v_1$ und sind paarweise disjunkt.

Bemerkung: Hier ist noch viel Potential:

- Man muss nur Bahnen gleicher Länge vergleichen.
- Für Disjunktheit zweier Bahnen muss nur eine ganz aufgezählt werden.
- Es muss also nur eine Bahn gleichzeitig in den Speicher passen.

Überblick und 2-modulare Charaktertafel von Fi_{23}

Wir haben hier

- eine Permutationsdarstellung von Fi_{23} auf 1 252 451 200 Punkten konstruiert,
- alle 36 N-Suborbits aufgezählt und die Längen bestimmt,
- N-N-Doppelnebenklassenvertreter $g_1, \ldots, g_{36}$ als Worte in a, b gefunden,
- damit die Voraussetzungen für Kondensationsrechnungen geschaffen und
- die 2-modulare Charaktertafel von Fi_{23} bestimmt (zusammen mit Gerhard Hiß und Felix Noeske).

Ausblick

Diese Methoden lassen sich anwenden, wenn

- geeignete (kleine) lineare Darstellungen verfügbar sind,
- ein geeigneter Vektor v_1 (oder Ähnliches) vorhanden ist,
- eine (oder mehrere) geeignete Hilfsuntergruppen U zu finden sind,
- große Bahnen bzw. Doppelnebenklassenvertreter gefragt sind.

Alles ist in GAP implementiert und wird in naher Zukunft als ein Paket veröffentlicht werden.