数理解析研究所講究録
941巻 1996年 73-83

10. Computational Group Theory への招待

藤本 光史 (富山工業高等専門学校)

10.1 Introduction

Computational Group Theory とは (i) 群の代数構造に関する性質を調べるアルゴリズムの開発、(ii) それらのアルゴリズムをインプリメントするシステムの開発、(iii) そのシステムを用いた有限群の分類などの応用に関する研究から成る分野である。現在、オーストラリアの Magma (Cayley の改良版)、ドイツの GAP などのシステムがあり、群論の専門家以外の人々にも広く利用されている。

ここでは Computational Group Theory の歴史と主要なアルゴリズムの紹介及び最近の研究状況について紹介する。

10.2 Elementary Problems

ここでは Computational Group Theory で扱う群のクラスや基本的な問題について述べる。

Definition 群 G の部分群の列

$$G = G_1 \geq G_2 \geq \cdots \geq G_n \geq G_{n+1} = 1$$

において、すべての i に対して G_{i+1} が G_i の正規部分群で、G_i/G_{i+1} が巡回群であるとき、この列を長さ n の *polycyclic series* という。polycyclic series をもつ群を *polycyclic group* という。

polycyclic group に関して次のような群の包含関係がある。

Fact

$$\{p\text{-groups}\}\subset\{\text{finite nilpotent groups}\}$$
$$\subset\{\text{finite soluble groups}\}\subset\{\text{polycyclic groups}\}\subset\{\text{soluble groups}\}$$

さらに polycyclic group について次の定理が成り立つ。

Theorem

(1) G is polycyclic group $\Longleftrightarrow$ G is soluble group and all subgroups are finitely generated.

(2) G is polycyclic with length n $\Longrightarrow$ G can be generated by n elements.

この定理より、polycyclic group はコンピュータ上で'扱いやすい'群と考えられる。Computational Group Theory では、特にこのクラスの群について盛んな研究が行われている。

次に、Computational Group Theory に関する問題で基本的なものを挙げる。

Problems

群 G が有限表示$< \mathcal{A} \mid \mathcal{R} >$で与えられると、誰もが考える基本的な問題は以下のものであろう。

(1) 有限群か?

(2) 位数はいくつか?

(3) 共約類は?

(4) どんな部分群をもつか?

(5) どんな正規部分群をもつか?

(5) 単純群か?

(6) 可換か?

(7) 巡回群か?

(8) ベキ零か?

(9) 可解か?

(10) どんな群構造をもつか?

etc

(1)~(8) はいくつかの有用なアルゴリズムが存在するが、(9),(10) は非常に困難な問題である。

10.3 History

有限群論の問題の一つとして、**The general problem for finite groups** と呼ばれるものがある。この問題には、コンピュータの応用が不可欠と早くから認識され、いくつかの重要なアルゴリズムが考え出された。以下では、この問題に関連して発展した Computational Group Theory の歴史について述べる。

1878 年···Cayley によって、次の The general problem for finite groups が提唱された ([4])。

正整数 N を与えたとき，位数 N の群をすべて決定せよ

1882 年から 1906 年にかけて位数 N が 4 個以下の素数の積で表されるときが、まず決定された。

1882 年…p^2, pq ([22])

1893 年…p^3, p^2q, pqr, p^4 ([13],[35])

1895 年…$pqrs$ ([14])

1899 年…p^3q ([34])

1902 年…p^2q^2 ([32])

1906 年…p^2qr ([8])

位数が 5 個以上の素数の積で表される場合については困難であったが、1930 年代前半までにこの問題は $128 = 2^7$ と $192 = 2^6 \cdot 3$ を除いて 215 までの位数について解かれた。

1930 年…位数 100 までの群の数のリスト作成 ([20])

1934 年…位数 101 から 215 までの群の数のリスト作成 ([18],[28],[29])

ここで、手計算によるこの問題へのアタックは下火になり、コンピュータの登場を待つことになる。しかし、この時期と前後して、群論におけるコンピュータの応用に非常に有効な二つのアルゴリズムが考え出された。

1934 年…P.Hall は、任意の word を normal word に変換する **Collection Algorithm** を与えた ([9])。

1936 年…Todd と Coxeter は、生成元と定義関係式によって表された群の位数を求める **Coset Enumeration** を与えた ([31])。

これらの群論アルゴリズムが考えられるようになったきっかけは、Dehn による次のような決定問題の提起によるところが大きい。

1911 年（M. Dehn）

群 G は有限表示$< \mathcal{A} \mid \mathcal{R} >$で与えられているとする。

Word Problem：$\mathcal{A}$ 上の任意の語 w に対して、$w = e$ in G であるかどうかを判定するアルゴリズムがあるか。

Conjugacy Problem：$\mathcal{A}$ 上の任意の語 u と v が G 上共役であるかどうかを判定するアルゴリズムがあるか。

Isomorphism Problem：任意の 2 つの群が同型であるかどうかを判定するアルゴリズムがあるか。

Dehn の提起以来、いろいろな群に関して、これらの決定問題が研究されている ([21])。

1946 年…ペンシルバニア大学でエッカートとモークリーが設計･構築したコンピュータ ENIAC が誕生する。

1953 年…Coset enumeration が Cambridge の B.Haselgrove によってはじめてコンピュータ上で実

行された ([16])。

1960 年···Collection アルゴリズムが H.Felsch によってはじめてコンピュータ上で実行された ([5])。

1967 年···研究集会 "Computational Problems in Abstruct Algebra" が Oxford で開かれた ([17])。

1970 年···L.Gerhards と E.Altmann は、有限可解群の自己同型群を決定するアルゴリズムを与えた ([7])。

1971 年···I.D.Macdonald は、任意の表示によって与えられた p-群の power-commutator presentation を生成する **Nilpotent Quotient Algorithm** を与え、自ら実行した ([19])。power-commutator presentation で与えられた群は、Word Problem が決定されるためこのアルゴリズムは p-群の研究に非常に有効なものであった。

1973 年···G.Havas は、群 G と部分群 H の G における指数 $|G:H|$ を与えると、H の群表示を求める **Reidemeister-Schreier program** を与えた ([10])。

1974 年···初の Computational Group Theory system "Aachen–Sydney Group System" が開発される。(**Cayley** の原型)

1975 年···M.F.Newman は、Nilpotent Quotient Algorithm を用いて、位数 p^n と生成元の数を与えると、その条件を満たす p-群を生成するp-**群生成アルゴリズム**の原型を与えた ([23],[24])。

1976 年···Alford, Ascione, Havas, Leedham-Green, Newman によって、p-群生成アルゴリズムの部分的実行が行われた ([1])。

1982 年···G.Havas、P.E.Kenne、J.S.Richardson 及び E.F. Robertson は、Reidemeister-Schreier program の部分的改良となる Tietze transformation program を与えた ([12])。

1982 年···研究集会 "**Computational Group Theory**" が Durhum で開催された ([2])。

1986 年···p-群生成アルゴリズムの全般的実行が Newman と O'Brien によって行われた。この実行は位数 128 の群は 2328 個あることを示した ([11],[15],[26])。

また、この年、Aachen の RWTH(Rheinisch–Westfälisches Technische Hochschule) で **GAP** が開発された。

1989 年···O'Brien によって位数 256 の群が 56092 個あることが示された ([27])。

1993 年···University of Sydney の Computational Algebra Group より **Magma**(version 1) がリリースされた。

10.4 Algorithms

10.4.1 Collection Algorithm

Definition 次のような群表示を *power-conjugate presentation* と呼ぶ。

$$< a_1, a_2, \cdots, a_n \mid a_i^{\rho(i)} = u_i, 1 \leq i \leq n,\ a_j a_i = a_i v_{ij}, 1 \leq i < j \leq n >$$

ただし、u_i, v_{ij} は $\{a_{i+1}, a_{i+2}, \cdots, a_n\}$ からなる word である。

上の表示を持つ群の位数が

$$\rho(1)\rho(2)\cdots\rho(n)$$

であるとき、この表示は *consistent* であるという。

与えられた power-conjugate presentation が consistent であるかどうかの判定には次の定理が用いられる。

Theorem(Wamsley [33])

power-conjugate presentation is consistent

$$\begin{aligned}
\Longleftrightarrow\ & (a_k(a_j a_i)) = ((a_k a_j)a_i) \quad \text{for } 1 \leq i < j < k \leq n, \\
& ((a_k^p)a_j) = (a_k^{p-1}(a_k a_j)) \quad \text{for } 1 \leq j < k \leq n, \\
& (a_j(a_i^p)) = ((a_j a_i)a_i^{p-1}) \quad \text{for } 1 \leq i < j \leq n, \\
& (a_i(a_i^p)) = ((a_i^p)a_i) \quad \text{for } 1 \leq i \leq n
\end{aligned}$$

Fact

(1) power-conjugate presentation で表された群は、位数が高々$\rho(1)\rho(2)\cdots\rho(n)$ の有限可解群である。

(2) 任意の有限可解群は、consistent power-conjugate presentation で表される。

(3) consistent power-conjugate presentation で表された有限可解群は Word Problem が決定されており、normal word $a_1^{k_1}a_2^{k_2}\cdots a_n^{k_n}$ $(0 \leq k_i < \rho(i))$ で元を一意的に表現できる。

任意の word を normal word に変換する Collection Algorithm を以下に示す ([3])。

Collection Algorithm

Input: a power-conjugate presentation with generators $\{a_1, a_2, \cdots, a_n\}$;

a word w in $\{a_1, a_2, \cdots, a_n\}$;

Output: a normal word w equivalent to the input word;

begin

while w is not normal **do**

w :=the result of replacing a subword of w which occurs

in list (A) by its equivalent in list(B);

end while;

end.

(A)	(B)
a_i^{-1}	$a_i^{\rho(i)-1}u_i^{-1}$
$a_i^{\rho(i)}$	u_i
$a_j a_i$	$a_i v_{ij}$

Example

位数 $18(=3^2\cdot 2)$ の非 Abel 群

$$< a_1, a_2, a_3 \mid a_1^3 = 1, a_2^3 = 1, a_3^2 = 1, a_2 a_1 a_2^{-1} = a_1, a_3 a_1 a_3^{-1} = a_2, a_3 a_2 a_3^{-1} = a_1 >$$

において、word $a_2^{-1}a_1^2a_3^{-1}a_1^{-1}a_2$ を normal word に変換する。

現在開発中の Prolog 群論計算システムでは以下のようになる。

```
?- r(R),get_normal_word([-2,1,1,-3,-1,2],R,NW).
[-2,1,1,-3,1,1,2]
[2,2,1,1,-3,1,1,2]
[2,2,1,1,3,1,1,2]
[2,1,2,1,3,1,1,2]
[1,2,2,1,3,1,1,2]
[1,2,1,2,3,1,1,2]
[1,1,2,2,3,1,1,2]
[1,1,2,2,2,3,1,2]
[1,1,3,1,2]
[1,1,2,3,2]
[1,1,2,1,3]
[1,1,1,2,3]
[2,3]
R = [[[1,1,1],[]],[[-1],[1,1]],[[2,2,2],[]],
[[-2],[2,2]],[[3,3],[]],[[-3],[3]],[[2,1],[1,2]],
[[3,1],[2,3]],[[3,2],[1,3]]]
NW = [2,3] ->
```

これによって $a_2^{-1}a_1^2a_3^{-1}a_1^{-1}a_2 = a_2a_3$ となることがわかる ([6])。

10.4.2 Consistent power-conjugate presentation を求めるアルゴリズム

任意の有限可解群は、consistent power-conjugate presentation を持つ。しかし、任意に与えられた群表示に対してその consistent power-conjugate presentation を求めることは容易なことではない。p-群に関しては、1971 年に Macdonald が考案した Nilpotent Quotient Algorithm によって、任意の表示で与えられた p-群を consistent power-conjugate presentation に変換できる。

Defnition 群 G, 素数 p に対して、

$$\mathcal{P}_0^p(G) = G, \quad \mathcal{P}_i^p(G) = [\mathcal{P}_{i-1}^p(G), G](\mathcal{P}_{i-1}^p(G))^p \quad (\text{for } i \geq 1)$$

とすると、部分群の列

$$G = \mathcal{P}_0^p(G) \geq \mathcal{P}_1^p(G) \geq \cdots \geq \mathcal{P}_i^p(G) \geq \cdots$$

をえる。この列を G の *lower exponent-p central series* という。

この列において、$\mathcal{P}_c^p(G) = 1$ となる最小の正整数 c があるとき、G は exponent-p class c をもつという。

Nilpotent Quotient Algorithm

Input: a finite presentation $< \mathcal{A} \mid \mathcal{R} >$ for G,

p: prime, c: non-negative integer

Output: consistent power-conjugate presentation for $G/\mathcal{P}_c^p$

特に、G が p-群のときは exponent-p class c をもつので $G/\mathcal{P}_c^p(G) = G$ となり、G 自身の consistent power-conjugate presentation が求められる。

有限可解群の consistent power-conjugate presentation を求めるアルゴリズムは **Finite Soluble Quotient Algorithm** と呼ばれ、現在まだなお研究中である。ここでは 1994 年に発表された Niemeyer による最新のアルゴリズムについて簡単に述べる ([25])。

Definition 素数 p_i, 非負整数 c_i の対から成るリスト $\mathcal{L} = [(p_1, c_1), (p_2, c_2), \cdots, (p_k, c_k)]$ （ただし $p_i \neq p_{i+1}$, また $i < k$では $c_i > 0$）に対して、

$$\mathcal{L}_{1,0}(G) = G,$$
$$\mathcal{L}_{i,j}(G) = \mathcal{P}_j^{p_i}(\mathcal{L}_{i,0}(G)) \quad \text{for } 1 \leq i \leq k, 1 \leq j \leq c_i,$$
$$\mathcal{L}_{i+1,0}(G) = \mathcal{L}_{i,c_i}(G)) \quad \text{for } 1 \leq i < k,$$
$$\mathcal{L}(G) = \mathcal{L}_{k,c_k}(G)$$

とすると、部分群の列

$$G = \mathcal{L}_{1,0}(G) \geq \mathcal{L}_{1,1}(G) \geq \mathcal{L}_{1,2}(G) \geq \cdots \geq \mathcal{L}_{1,c_1-1}(G) \geq \mathcal{L}_{1,c_1}(G)$$
$$= \mathcal{L}_{2,0}(G) \geq \mathcal{L}_{2,1}(G) \geq \mathcal{L}_{2,2}(G) \geq \cdots \geq \mathcal{L}_{2,c_2-1}(G) \geq \mathcal{L}_{2,c_2}(G)$$
$$\cdots\cdots\cdots$$
$$= \mathcal{L}_{k,0}(G) \geq \mathcal{L}_{k,1}(G) \geq \mathcal{L}_{k,2}(G) \geq \cdots \geq \mathcal{L}_{k,c_k-1}(G) \geq \mathcal{L}_{k,c_k}(G) = \mathcal{L}(G)$$

をえる。この列を G の $\mathcal{L}$-*series* という。この列において $\mathcal{L}(G) = 1$ となるとき、G は $\mathcal{L}$-*group* と呼ばれる。さらに、$c_k > 0$ かつ $\mathcal{L}_{k,c_k-1}(G) \neq 1$ であるとき、G は *strict* $\mathcal{L}$-*group* と呼ばれる。

Note

$$\mathcal{L}_{i,j}(G) = \mathcal{P}_j^{p_i}(\mathcal{P}_{c_{i-1}}^{p_{i-1}}(\mathcal{P}_{c_{i-2}}^{p_{i-2}}(\cdots(\mathcal{P}_{c_1}^{p_1}(G))\cdots)))$$

Fact

任意の有限可解群に対して、strict $\mathcal{L}$-group となるリスト $\mathcal{L}$ が存在する。

Finite Soluble Quotient Algorithm

Input: a finite presentation $< \mathcal{A} \mid \mathcal{R} >$ for G,

$\mathcal{L} = [(p_1, c_1), (p_2, c_2), \cdots, (p_k, c_k)]$

Output: consistent power-conjugate presentation for $G/\mathcal{L}$

Examples

Presentation	$\mathcal{L}$	Order
$< a, b \mid a^3, b^6, (ab)^6, (a^{-1}b)^6 >$	$[(3,2),(2,2)]$	$2^{182}\cdot 3^3$
$< a, b \mid (ab)^2 b^{-6}, a^4 b^{-1} ab^{-9} a^{-1} b >$	$[(2,1),(3,1),(2,2),(3,2)]$	$2^4\cdot 3^4$
$< a, b \mid ab^2 (ab^{-1})^2, (a^2 b)^2 a^{-1} b a^2 (bab)^{-1} >$	$[(3,1),(2,2),(5,2)]$	$2^3\cdot 3\cdot 5^3$
$< a, b \mid ab^2 a^{-1} b^{-1} a^3 b^{-1}, ba^2 b^{-1} a^{-1} b^3 a^{-1} b^3 a^{-1} >$	$[(3,2),(2,2)]$	$2^8\cdot 3^3$
$< a, b \mid a^8, b^3, (a^{-1}b)^2, (ba^3 ba)^2 a^{-4} >$	$[(2,1),(3,1),(2,1)]$	$2^3\cdot 3$

一番下の例は 4 次対称群 S_4 である。GAP の最新バージョン (version 3 release 4) では ANUSQ という名称の Share Library をインストールすることによって、Finite Soluble Algorithm が利用できるようになった。GAP を使って S_4 の consistent power-conjugate presentation を求めてみる。

```
gap> RequirePackage("anusq");
gap> f:=FreeGroup("a","b");
Group( a, b )
gap> f:=f/[f.1^8,f.2^3,(f.1^-1*f.2)^2,(f.2*f.1^3*f.2*f.1)^2*f.1^-4];
Group( a, b )
gap> Size(f);
24
gap> g:=Sq(f,[[2,1],[3,1],[2,1]]);
```

```
rec(
  generators := [ a.1, a.2, a.3, a.4 ],
  relators := [ a.1^2*a.3^-1, a.1^-1*a.2*a.1*a.3^-1*a.2^-2, a.2^3,
      a.1^-1*a.3*a.1*a.3^-1, a.2^-1*a.3*a.2*a.4^-1, a.3^2,
      a.1^-1*a.4*a.1*a.4^-1*a.3^-1, a.2^-1*a.4*a.2*a.4^-1*a.3^-1,
      a.3^-1*a.4*a.3*a.4^-1, a.4^2 ] )
gap>
```

Open Problem

現在、次の問題がまだ未解決である。

有限可解群 G に対して、$\mathcal{L}(G) = 1$ となるリスト $\mathcal{L}$ を求める

10.5 文献について

これから Computational Group Theory を勉強したいという方には、最近出版された Sims の本([30]) を読まれることをお薦めする。また、この分野に関する文献リスト（BibTeX 形式）が次のサーバーから anonymous ftp により入手できるので、活用していただきたい。

```
pell.anu.edu.au (IP 150.203.33.4) :/pub/bib/algebra.bib
```

GAP の最新版については以下のサーバーから同じく anonymous ftp により入手できる。

```
samson.math.rwth-aachen.de (IP 137.226.152.6) :/pub/gap/gap3r4p2.zoo
```

参考文献

[1]Judith A. Ascione, George Havas and C. R. Leedham-Green, A computer aided classification of certain groups of prime order, *Bull. Austral. Math. Soc.* 17(1977), 257–274.

[2](ed. by) M. D. Atkinson, *Computational Group Theory*, Academic Press, London/New York, 1984.

[3]G. Butler, *Fundamental Algorithms for Permutation Groups*, (Lecture Notes in Computer Science 559) Springer-Verlag, Berlin/Heidelberg/New York, 1991.

[4]A. Cayley, Desiderata and suggestions. No.1. The theory of groups, *American Journal of Mathematics,* 1(1878), 50–52.

[5]H. Felsch, Die Behandlung zweier gruppen- theoretischer Verfahren auf elektronischen Rechenmaschinen, Diplomarbeit, Kiel, 1960.

[6]藤本 光史, Prolog による群論計算システム (2), 数式処理 (*to appear*).

[7]L. Gerhards and E. Altmann, A computational method for determinating the automorphism group of a finite solvable group, *Computational Problems in Abstract Algebra* (ed. by J. Leech), Pergamon Press, Oxford, 1970, 61–74.

[8]O. E. Glenn, Determination of the abstract groups of order p^2qr, *Transactions of the American Mathematical Society*, vol. 7(1906), 137–151.

[9]P. Hall, A contribution to the theory of groups of prime-power order, *Proceedings of the London Mathematical Society (2),* 36(1934), 29–95.

[10]G. Havas, A Reidemeister-Schreier program, *The Theory of Groups* (*Canberra 1973*), (Proc. second internat. conf. The theory of groups, Lecture Notes in Mathematics, vol. 372) Springer-Verlag, Berlin/Heidelberg/New York, 1974, 347–356.

[11]G. Havas and M. F. Newman, Application of computers to questions like those of Burnside, *Burnside Groups* (*Bielefeld, 1977*), (Lecture Notes in Mathematics, vol. 806) Springer-Verlag, Berlin/Heidelberg/New York, 1980, 211–230.

[12]G. Havas, P. E. Kenne, J. S. Richardson, and E. F. Robertson, A Tietze transformation program, *Computational Group Theory* (*Durham, 1982*), (ed. by Michael Atkinson), Academic Press, London/New York, 1984, 69–73.

[13]Hölder, Die Gruppen der Ordnungen p^3, p^2q, pqr, p^4, *Mathematische Annalen,* 43(1893), 301–412.

[14]Hölder, Die Gruppen mit quadratfreier Ordnungszahl, *Abh. Akad. Wissenschaften zu Göttingen Mathematisch-physikalisch Klasse* (1895), 211–229.

[15]Rodney James, M. F. Newman and E. A. O'Brien, The groups of order 128, *Journal of Algebra* 129(1990), 136–158.

[16]J. Leech, Coset enumeration on digital computers, *Proceedings of the Cambridge Philosophical Society*, 59(1963), 257–267.

[17](ed. by) J. Leech, *Computational Problems in Abstract Algebra*, Pergamon Press, Oxford, 1970.

[18]A. C. Lunn and J. K. Senior, A method of determining all the solvable groups of given order and its application to the orders $16p$ and $32p$, *American Journal of Mathematics,* 56(1934), 319–327.

[19]I. D. Macdonald, A computer application to finite p-groups, *Journal of the Australian Mathematical Society* 17(1974), 102–112.

[20]G. A. Miller, Determination of all the groups of order 64, *American Journal of Mathematics,* 52(1930), 617–634.

[21]C. F. Miller, Decision Problems for Groups - Survey and Reflections, *Algorithms and Classifi-*

cation in Combinatorial Group Theory, Springer-Verlag, Berlin/Heidelberg/New York, 1992, 1–59.

[22]E. Netto, *Substitutionentheorie und ihre Anwendungen auf die Algebra*, Teubner, Leipzig, 1882.

[23]M. F. Newman, Determination of groups of prime-power order, *Group Theory(Canberra 1975)*, (Proc. Miniconf. Australian National University, 1975. Lecture Notes in Mathematics, vol. 573) Springer-Verlag, Berlin/Heidelberg/New York, 1977, 73–84.

[24]M. F. Newman, Calculating presentations for certain kinds of quotient groups, *SYMSAC'76 (Proc. 1976 ACM Symp. on Symbolic and Algebraic Computation, Yorktown Heights, 1976)*, ed. by R. D. Jenks, ACM, New York, 1976, 2–8.

[25]A. C. Niemeyer, A Finite Soluble Quotient Algorithm, *J. Symbolic Computation* 18(1994), 541–561.

[26]E. A. O'Brien, The p-group generation algorithm, *J. Symbolic Computation* 9(1990), 677–698.

[27]E. A. O'Brien, The groups of order 256, *J. Algebra* 143(1991), 219–235.

[28]J. K. Senior and A. C. Lunn, Determination of the groups of orders 101–161, omitting order 128, *American Journal of Mathematics,* 56(1934), 328–338.

[29]J. K. Senior and A. C. Lunn, Determination of the groups of orders 162–215, omitting order 192, *American Journal of Mathematics,* 57(1935), 254–260.

[30]C. C. Sims, *Computation with finitely presented groups*, Cambridge University Press, 1994.

[31]J. A. Todd and H. S. M. Coxeter, A practical method for enumerating cosets of a finite abstract group, *Proceedings of Edinburgh Mathematical Society,* 5(1936), 26–34.

[32]Le Vavasseur, Les groupes d'ordre p^2q^2, p étant un nombre premier plus grand que le nombre premier q, *Annales scientifiques de l'Ecole Normale Supérieure (3),* 19(1902), 335–355.

[33]Wamsley, Computing soluble groups, *Group Theory(Canberra 1975)*, (Proc. Miniconf. Australian National University, 1975. Lecture Notes in Mathematics, vol. 573) Springer-Verlag, Berlin/Heidelberg/New York, 1977, 118–125.

[34]A. E. Western, Groups of order p^3q, *Proceedings of the London Mathematical Society (1),* 30 (1899), 209–263.

[35]J. W. A. Young, On the determination of groups whose order is a power of a prime, *American Journal of Mathematics,* 15(1893), 124–178.