Chapter 3

Computational Group Theory

In this introductory chapter we give a brief survey of some results from computational group theory. Our main focus will be on the complexity of group-theoretic problems. We are interested when the groups involved in the problem instances are specified by generator sets.

Research in computational group theory centers mainly on problems concerning *permutation groups* and *matrix groups*. Among the basic problems that have been studied are membership testing, order computation, normal closure computation, computing structural elements like composition series, center, Sylow subgroups etc[1].

Much of the research in the area has gone into developing efficient algorithms for permutation groups (these are the subgroups of the symmetric group). For this class of groups many computational problems have polynomial time solutions. Central in many of these algorithms is the notion of a *strong generator set* for a permutation group introduced by C. Sim [Sim70]. The importance of this notion is mainly because of the fact that many basic problems like membership testing, order computation and computing the normal closure are efficiently reducible to the computation of a strong generator set. Sim [Sim70] gave a method for constructing a strong generator set from an arbitrary generator set. A variant of Sim's algorithm was shown to be in

[1] We omit the formal definitions of these problems. Please see [KL90] for definitions of these problems in the context of permutation groups.

polynomial time by Furst et. al. in [FHL80]. This gives polynomial-time algorithms for membership testing, order computation and computing the normal closure.

The problem of computing the composition series and center for permutation groups was shown to be in polynomial time by E. Luks [Luk87]. The analysis of Luks' algorithm uses many deep permutation group-theoretic results along with a detailed knowledge of the classification of finite simple groups. The problem of computing a generator set of a Sylow p-subgroup for prime p, was shown to be in polynomial time by W. Kantor [Kan85]. A library of polynomial time algorithms for a long list of problems over permutation groups (even for factor groups) is given in [KL90]. In [BLS87], Babai et. al. have shown that membership testing, order computation, computing the center and the composition series can be performed in NC. These results also depend on consequences of the classification of finite simple groups for their analysis.

Though there are efficient algorithms for a large number of problems over permutation groups, the scenario is quite different in the case of matrix groups over finite fields (these are subgroups of the group of invertible matrices over a finite field). The problem of testing membership for 1×1 matrix groups over finite fields is the decision version of the discrete logarithm problem, for which no polynomial time algorithm is known. The techniques developed for fast management of permutation groups will not work for matrix groups. The main reason for this is that, while permutation groups act on a set of "small" size (the subgroups of S_n act on a set of size n), the action of $n \times n$ invertible matrices over a finite field is on a vector space of size *exponential* in n. The polynomial-time algorithms for testing solvability and nilpotence of matrix groups over finite fields, due to Luks [Luk92], is an important result in this area.

From the point of view of complexity theory, these problems over matrix groups are interesting since the exact complexity of many of these problems are not characterized. In order to study the complexity of these hard problems in a general setting

Babai and Szemerédi [BS84] introduced the notion of black-box groups. The main motivation was to avoid the the actual representation of group elements while investigating the complexity of group-theoretic problems. Intuitively in this framework, the group elements are uniformly encoded as binary strings and the group operations are performed by a group oracle. Hence an upper bound for a computational problem in this framework gives the same upper bound for the problem over groups for which the operations can be done efficiently.

Let us consider the complexity of testing membership in black-box groups. Notice that it is not obvious whether the problem is in NP. Naive idea of guessing a product over generators does not work. But in [BS84] it is shown that for any finite group G, an element of G can be "reached" using a short straight line code over an arbitrary generator set of G. From this reachability result it easily follows that membership testing over general black-box groups is in NP.

Verification of the order (decision version of computing the order) is a central problem when we are interested in the complexity of group theoretic problems; many problems reduce to order verification [Bab92]. For example, it is easy to see that both membership and non-membership testing reduces to verification of order. For matrix groups over finite fields, it is believed that order verification is in NP. In [BS84] it is shown that, over solvable matrix groups, order verification in NP $\cap$ co-NP. This puts membership testing in NP $\cap$ co-NP for this class of groups. This is a strong indication that membership testing over solvable matrix groups cannot be complete for NP (unless NP = co-NP).

In [Bab92], using sophisticated combinatorial and group-theoretic techniques, it is shown that order verification over general black-box groups is in the class AM $\cap$ co-AM [Bab85]. This places membership testing over general black-box groups in NP$\cap$co-AM. Since AM$\cap$co-AM is known to be low for the second level of PH [Sch88], it is unlikely that the problems membership testing and order verification are hard for NP. In [Bab92], using this above-mentioned upper bound result, similar upper bounds were shown for a large number of problems over black-box groups. For example, it is shown that isomorphism testing over black-box groups is in AM.

Randomization has also played an important role in computational group theory. The notion of *random normal subproduct* introduced in [BCF+95] has given rise to an efficient randomized algorithm for computing normal closures over black-box groups. This, along with random subproducts [BCF+95] gives a co-RP algorithm for testing solvability (see also [CF93]).

There are computationally difficult problems concerning permutation groups also. Group intersection, group factorization and coset intersection are some among them (see [KST92] for exact definitions of these problems). There are no polynomial time algorithms known for these problems. On the other hand these problems are in NP $\cap$ co-AM [Bab92] and hence are not complete for NP unless the polynomial-hierarchy collapses. The counting complexity of these problems were first investigated in [KST92]. It is shown in [KST92] that group intersection is in SPP while group factorization and coset intersection (along with Graph Isomorphism), belongs to LWPP. Hence these problems are low for the counting classes PP and $C_=P$. These results shows the similarity of the complexity of these problems to that of Graph Isomorphism. There are also some problems over permutation groups that are NP-complete. One interesting example is the problem of deciding whether a group contains an element which is fixed-point free [Lub81].

In the next chapter we investigate the counting complexity of three basic problems; membership testing, order verification and isomorphism testing, over black-box groups. We show that over solvable black-box groups these problems are in SPP. This result has appeared in [AV97b, Vin97]. In [AV97b, Vin97], it is also shown that over solvable black-box groups, the problems group intersection, group factorization and coset intersection are in LWPP.