1. Introduction

Numerous problems in the mathematical and physical sciences can be reduced to questions of counting solutions in combinatorial structures. Much effort has been put into developing analytic techniques for doing this effectively for the various problems that arise most frequently. A glance at the literature, however, suggests that the search for positive results has had only very limited success, and that for the majority of questions we still cannot count exactly in any effective sense.

In this paper we aim to survey recent techniques that enable negative results to be proved. Three classes of problems are defined that contain many of the best known problems. On historical grounds membership in each of these classes can be interpreted as overwhelming evidence of intrinsic intractability, in the same sense as NP-completeness for combinatorial search problems. In some areas we are able to classify most of the previously open problems as intractable in this sense, thereby confirming that the known positive techniques go as far as they can. In other cases, however, the resulting classification leaves the large gaps in our knowledge virtually untouched.

For uniformity we shall restrict ourselves to graph-theoretic problems. Other discrete structures, such as Boolean functions, can be treated similarly.

We shall assume that graphs are undirected, except where otherwise stated. The particular problems on which we focus are the following. The dimer problem is that of counting the number of perfect matchings in a graph. It is motivated by its several applications in the physical sciences [12,19]. The monomer-dimer problem counts matchings of arbitrary size and is similarly motivated. Self-avoiding walks are paths that do not go through any node more than once. Discussions of applications are given by Frisch and Hammersley [10] and Barber and Ninham [4]. A self-avoiding walk that goes through every node is a Hamiltonian path. Counting spanning trees first appeared in connection with electrical flow (Kirchhoff [20]), and of unlabelled trees in chemistry (Cayley [8]). An Eulerian path is one that contains every edge exactly once. Counting the number of subgraphs of a graph that connect a given pair of nodes corresponds to a reliability problem. In regular lattice graphs varieties of it appear as percolation problems [10,32]. A complete subgraph is a maximal clique if it is a complete subgraph that is not properly contained in a larger one. It is a k-clique if it has cardinality k.

Our first two notions of intractability, \#P-completeness and \#P_1-completeness, concern the runtime of discrete computations and the non-existence of algorithms
that run in polynomial time. The third notion, completeness over field $F$, is an algebraic one. It refers to the generating polynomials associated with the counting problem. Its implications include not only the complexity of algebraic programs in the sense of [6,p6] but also, we believe, the nonexistence of analytic techniques for solving these problems within a certain class [30]. The three classes will be defined informally in the next section. Examples of their members will be given in sections 3, 4 and 5.

The three notions correspond to the different levels of effectiveness at which we may hope to count substructures in a graph. Suppose graph $G$ is defined by the pair $(V,E)$ when $V$ is the set of nodes, and $E = \{e_1, \ldots, e_r\}$ the set of edges. Let $M = \{E_1, E_2, \ldots, E_m\}$ be the set of subsets of $E$ that correspond to the occurrences of the substructures (e.g. $M$ may be the set of perfect matchings or the set of self-avoiding walks.) Then the full polynomial for $M$ over field $F$ is

$$
\sum_{j=1}^{m} \prod_{e \in E_j} x_e
$$

regarded as a polynomial over indeterminates $\{x_1, \ldots, x_r\}$ with coefficients from the field $F$. If this polynomial can be computed and manipulated easily then we are in an excellent position not only to count the number of solutions, by substituting the identity for the indeterminates, but also to get further information, such as the asymptotic behaviour. If this polynomial is not easily computable itself there may still be a fast discrete algorithm for the counting problem itself. A third still weaker situation arises when such algorithms do not exist for arbitrary graphs but do for one special graph of each size. For example if we want to know how many labelled graphs on $n$ nodes have some property, then we can rephrase this by asking how many subgraphs of the complete graph $K_n$ on $n$ nodes have the property. A second common example is the restriction of the graph to rectangular lattice grids, as is often sufficient for physical applications. When none of the three levels of counting is possible we may hope for approximate techniques, or for asymptotic results. For many of the structures defined above even these questions are problematic although much research has been done on them.

In section 6 an application of counting in computer science is discussed. In the final section brief mention is made of the problems that arise in counting when we wish to identify classes of objects according to some equivalence relation, typically isomorphism. This is often referred to as unlabelled enumeration to distinguish it from labelled enumeration which is the subject of the other sections.

2. Definitions

For discrete computations we assume that all objects are encoded over finite alphabets $Z, I$. The size of an object $x$ is $|x|$, the number of symbols required to represent it. Suppose $X \subseteq Z^*$ and $Y \subseteq I^*$ are sets of encoded objects. A