1 Introduction

Optimization problems appear in many situations in our daily life. If you, e. g., connect to the web server of a train company or travel agency, you can search connections between far-distant places for optimal travel times, ticket prices or number of changes, etc. In a more general economic context, such problems appear whenever a process has to be arranged in such a way that resources, money, or time are saved. Also in science optimization problems are of central interest, e. g., in physics for understanding the low-temperature behavior of model systems, or in biology for extracting information out of a huge amount of experimental data.

Probably you first encountered optimization in school: In mathematics courses one-dimensional functions over the space of real variables are analyzed, including the determination of minima and maxima. Such problems are in general easily solvable – at least

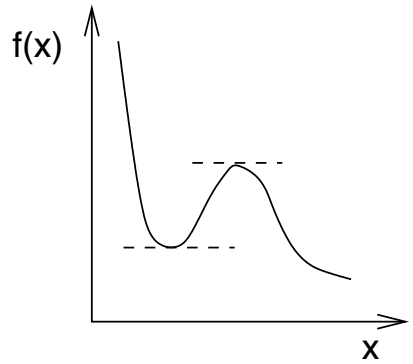

Figure 1.1: Extremal points of a function defined over a one-dimensional continuous space.

on a computer. In this book we are going to discuss so-called combinatorial optimization problems. These are in practice much harder to solve. First, the function to be minimized is in general high-dimensional; second, the variables are of discrete nature, and methods of continuous analysis such as taking derivatives do not work.

In this introductory chapter, we will present two examples which give you a flavor of combinatorial optimization. Furthermore, we will explain why optimization problems are interesting for physicists and how statistical mechanics can contribute to the understanding and solution of them. Finally, we recommend and discuss some basic textbooks related to the field.

Phase Transitions in Combinatorial Optimization Problems. Alexander K. Hartmann and Martin Weigt
Copyright © 2005 Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim
ISBN: 3-527-40473-2

1.1 Two examples of combinatorial optimization

Let us start by briefly formalizing the above explanation of combinatorial optimization problems. In general they are defined over some high-dimensional space, we will consider multi-component variables $\underline{\sigma} = (\sigma_1, \ldots, \sigma_n) \in X^n$ with $n \gg 1$. In addition, the problem is characterized by some *cost function* $H : X^n \to \mathbb{R}$ which assigns some cost to each $\underline{\sigma}$. This cost has to be minimized, leading to the following definition:

Definition: *minimization problem*

Given an n-dimensional space X^n and a cost function $H : X^n \to \mathbb{R}$.
The *minimization problem* reads as follows:

$$\textit{Find } \underline{\sigma}^{(0)} \in X^n \textit{, such that } H(\underline{\sigma}) \geq H(\underline{\sigma}^{(0)}) \textit{ for all } \underline{\sigma} \in X^n$$

The vector $\underline{\sigma}^{(0)}$ is called a *solution* of the minimization problem.

Finding a maximum is as easy or as hard as finding a minimum, since $\max H = -\min(-H)$. In addition we speak of *combinatorial* problems if the components of $\underline{\sigma}$ are of *discrete* nature, frequently used examples are $X = \{-1, 1\}$ (Ising spins), $X = \{true, false\}$ (Boolean variables) or $X = \mathbb{Z}$ (integer numbers).

Combinatorial optimization problems can be additionally complicated by the presence of *constraints*. The constraints reduce the size of the search space. At first glance this seems to facilitate the search for an optimal solution. The opposite is, however, frequently the case: Many optimization problems which can be solved efficiently on a computer without constraints, become extremely computer-time consuming if constraints are added.

To illustrate these rather abstract concepts, we will present two examples. The first one is a classical optimization problem in computer science, originating in economics.

Example: Traveling Salesman Problem (TSP)

Consider n towns distributed in a plane, numbered by $1, \ldots, n$. A salesman wants to visit all these towns, and at the end of his travels he wants to come back to his home town. He is confronted with the following minimization task: He has to find the shortest round-tour visiting every town exactly once. The problem is thus described by

$$\begin{aligned} X &= \{1, 2, \ldots, n\} \\ H(\underline{\sigma}) &= \sum_{i=1}^{n} d(\sigma_i, \sigma_{i+1}) \end{aligned} \tag{1.1}$$

where $d(\sigma_i, \sigma_j)$ is the distance between the two towns σ_i and σ_j, and $\sigma_{n+1} \equiv \sigma_1$ are identified with each other. The constraint that every town is visited once and

only once can be realized by constraining the vector $\underline{\sigma}$ to be a permutation of the sequence $(1, 2, \ldots, n)$.

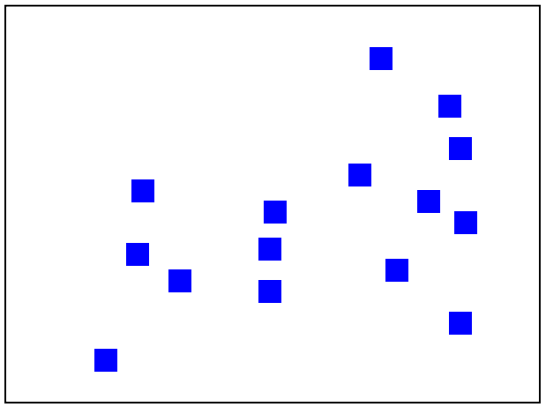

Figure 1.2: 15 cities in a plane.

As an example, 15 towns in a plane are given in Fig. 1.2; and one can try to find the shortest tour. For the general TSP the cities are not necessarily placed in a plane, but an arbitrary distance matrix d is given. □

The TSP is a so-called *hard* optimization problem. In this context, hardness is measured by the time a computer needs to solve this problem numerically, and a problem is considered to be hard if this time grows exponentially (or even faster) with the number n of components of the variable $\underline{\sigma}$. In TSP, one out of $(n-1)!$ round-tours has to be selected, and so far no good general selection criterion exists. Note that TSP has attracted not only computer scientists, but also physicists [1–6].

For an arbitrary algorithm, to describe the dependence between a suitably chosen measure n of the problem size and the running time T, the $\mathcal{O}$ notation is used.

Definition: *$\mathcal{O}$ notation*

Let $T, g : \mathbb{N} \to \mathbb{R}$ be two real-valued functions.
We write $T(n) = \mathcal{O}(g(n))$, iff there exists a positive number $c > 0$, such that $T(n) \leq cg(n)$ is valid for all $n > 0$. We say, $T(n)$ is *of order* at most $g(n)$.

Since constants are ignored when using the $\mathcal{O}$ notation, one speaks of the *asymptotic* running time or *time complexity*. In theoretical computer since, usually on states an upper bound over all possible inputs of size n, i. e., the *worst-case running time*. In this book, we will also study *typical* running times regarding a given ensemble of instances, see Sec. 1.2.

In Table 1.1, orders of running times, which occur typically in the context of algorithms, are presented, accompanied by the resulting values for problem sizes 10, 100, and 1000.

Table 1.1: Growth of functions as a function of input size n.

$T(n)$	$T(10)$	$T(100)$	$T(1000)$
n	10	100	1000
$n \log n$	10	200	3000
n^2	10^2	10^4	10^6
n^3	10^3	10^6	10^9
$n^{\log n}$	10	10^4	10^9
2^n	1024	1.3×10^{30}	1.1×10^{301}
$n!$	3.6×10^6	10^{158}	4×10^{2567}

Usually one considers problems as *easy*, if the running time is bounded by a polynomial, all others are considered as *hard*. The reason can be understood from the table: Even if the polynomial functions may take higher values for small n, asymptotically non-polynomial functions diverge much faster. Let us consider, e. g., the relative performance of two computers, one being twice as fast as the other one. In a linear-time problem, the faster computer is able to solve a problem which is twice as large as the problem solvable in the same time on the slower computer. If the running time grows, however, as 2^n, the faster computer is just able to go from n to $n + 1$ compared with the slower one. We see that for such hard problems, the utility of higher-speed computers is very limited – a substantial increase in the size of solvable problems can only be achieved via the use of better algorithms.

Often it is not immediately obvious whether or not a problem is easy. While finding the shortest round tour in the TSP is hard, finding the shortest path between two given towns (possibly through other cities) is easy, as we will see in Sec. 11.4.

Also in physics, many problems either are, or can be translated into, optimization problems. Examples are the determination of ground states of magnetic systems, the calculation of the structure of a folded protein, the analysis of data, or the study of flux lines in superconductors. This will be illustrated using a classical model in statistical physics.

Example: Ising spin glasses

Spin glasses [7, 8] are amorphous magnetic materials. The atoms in the solid carry microscopically small magnetic moments, which interact via couplings, some are ferromagnetic while others are antiferromagnetic. Due to the amorphous nature of the material, these couplings are disordered, i. e., they do not have any periodic structure. Spin glasses show an interesting frozen low-temperature phase which is, despite ongoing research over more than three decades, still poorly understood. Spin glasses can be modeled in the following way:

The magnetic moments are described by *Ising spins* σ_i which, due to anisotropies of the material, can take only two orientations called *up* and *down*, mathematically formalized by $\sigma_i = \pm 1$. In the simplest model one assumes that these spins are

placed on the sites of a regular lattice and that a spin interacts only with its nearest neighbors. The model is thus described by

$$\begin{aligned} X &= \{-1, 1\} \\ H(\underline{\sigma}) &= -\sum_{<i,j>} J_{ij}\sigma_i\sigma_j \end{aligned} \tag{1.2}$$

where J_{ij} denotes the interaction strength between the spins on sites i and j, and the sum runs over all pairs $\langle i, j\rangle$ of nearest neighbors. The function H measures the total energy of the system, and is called the *Hamiltonian*. Note that the interaction parameters are fixed, they are also called *quenched* variables, whereas the Ising spins are subject to thermal fluctuations and my change their values.

In a *ferromagnet* it is energetically favorable for any two neighboring spins to assume equal orientations $\sigma_i = \sigma_j$, i. e., all J_{ij} are positive, and parallel spins lead to a lower contribution to the total energy than antiparallel ones. So the system tends to be globally oriented in the same way. On the other hand, thermal noise causes spins to fluctuate randomly. At low temperatures T this thermal noise is small, and energetic contributions dominate over random spin fluctuations. The system becomes globally *ordered*. For temperatures higher than some critical temperature T_c, this long-range order becomes destroyed; a *phase transition* occurs at T_c. In Chap. 5, this phenomenon will be discussed in more detail in the context of a short introduction to statistical mechanics.

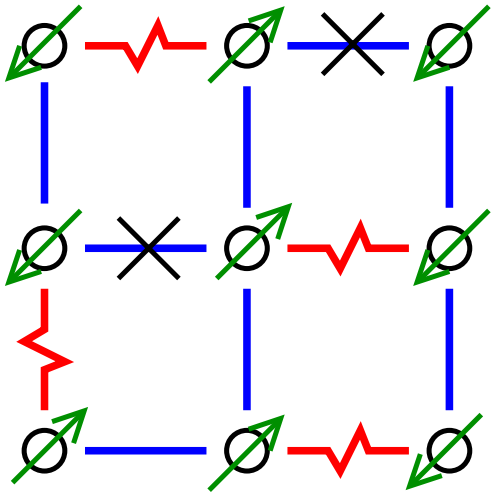

Figure 1.3: Two-dimensional spin glass. Solid lines represent ferromagnetic interactions, while jagged lines correspond to antiferromagnetic interactions. The small arrows represent the spins, adjusted to a ground-state configuration. For all except two interactions (marked by the crosses) the spins are oriented relative to each other in an energetically favorable way. It is not possible to find a state with lower energy.

At temperature $T = 0$ thermal fluctuations are completely absent, and the energy $H(\underline{\sigma})$ of the system assumes its global minimum. The corresponding spin configuration is called a *ground state*. The properties of these configurations are of great interest in physics, they serve as a basis for understanding the low-temperature behavior of physical systems. In the case of a ferromagnet these ground states are, as discussed above, extremely simple; all spins have the same orientation.

This becomes more complicated for *spin glasses* because ferromagnetic interactions ($J_{ij} > 0$) coexist with *antiferromagnetic* ones ($J_{ij} < 0$). Two spins connected by such an antiferromagnetic coupling prefer energetically to have opposite orientations. As already noted, spin glasses are amorphous materials, therefore ferromagnetic and antiferromagnetic interactions are distributed randomly on the bonds of the lattice. Consequently, it is not obvious how the ground state configurations of $H(\underline{\sigma})$ appear, and determining the minimum energy becomes a non-trivial minimization problem.

Figure 1.3 shows a small two-dimensional spin glass and one of its ground states. For this type of system usually many different ground states are feasible for each realization of the disorder. One says, the ground state is *degenerate*. Algorithms for calculating degenerate spin-glass ground states are explained in Chap. 11. As we will explain there, this calculation is easy (polynomial running time) for two-dimensional systems, but it becomes hard (exponential running time) in three dimensions. □

1.2 Why study combinatorial optimization using statistical physics?

The interdisciplinary exchange between computer science and statistical physics goes in both directions:

On one hand, many problems in daily life and in many scientific disciplines can be formulated as optimization problems. Hence, progress in the theory of combinatorial optimization, and in particular the development of new and more efficient algorithms influences many fields of science, technology and economy. In this obvious sense, computer science contributes to the understanding of physical systems. We will give a short introduction on the application of optimization algorithms for physics problems in Chap. 11.

On the other hand, physics can also help to shed light onto some basic, yet unsolved, questions in computer science. In statistical mechanics, many analytical and numerical methods have been developed in order to understand the macroscopic thermodynamic behavior of models starting from some microscopic description of a system via its Hamiltonian. These methods can be reinterpreted from the point of view of optimization theory. Problems, which are originally formulated as purely combinatorial tasks, can be equivalently rewritten as physical

models, by identifying the cost function with a Hamiltonian. Applying statistical mechanics tools at a temperature close to $T = 0$, may thus unveil many properties of the original problem and its cost-function minima; this will be the main focus of this book. In this way we will obtain insight into the intrinsic reasons for the hardness of these problems. For example, we will see that the hardness of many optimization problem is closely related to the *glassiness* of the corresponding physical system. Using this alternative physical approach, many interesting results have already been obtained, which have not been found before by applying traditional methods from mathematics and computer science.

Furthermore, traditional computer science defines the hardness according to a *worst-case* scenario. People dealing with practical applications are, however, more interested in *typical* instances of an optimization task rather than looking for the hardest possible instances. For this reason, suitably parametrized random ensembles of instances of problems have been introduced over recent years. In this context, it was observed that in some regions of the ensemble space instances are *typically* easy to solve, i. e., in a polynomially increasing running time, while in other regions instances are found to be typically hard. This change in behavior resembles the phase transitions observed in physical systems, like spin glasses or other disordered materials. Once more it is tempting to exploit the long experience of statistical physics with phase transitions in order to understand this behavior.

Example: Phase transitions in the TSP

As an example, we consider again the TSP. We study the random ensemble, where a plane of area size $A = L_x \times L_y$ is given. Each random instance consists of n cities, which are randomly placed in the plane, with $(x_i, y_i) \in [0, L_x] \times [0, L_y]$ denoting the position of city i. All positions are equally probable. The distances between pairs of cities are just Euclidean distances, i. e., $d(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

For each random instance of the problem, we ask the question:

Is the shortest round trip through all cities shorter than a given length l?

To answer this question for a given instance, one can use a *branch-and-bound* algorithm. For an introduction to this type of algorithm, see Sec. 6.3. Here, the algorithm basically enumerates all possible permutations of n cities stepwise, by selecting one city after the other. Some efficient heuristic is used, which determines the order in which the cities are tried, e. g., one starts the construction of the round trip, by first taking the two cities which are closest to each other, for details see Ref. [9]. The algorithm furthermore maintains a lower bound $l_{\min}$ of the tour length, determined by the cities selected so far. If $l_{\min} > l$, then the algorithm does not have to pursue permutations containing the same order of the cities selected so far, and it can search in other regions of the permutation space. On the other hand, if a full round trip with $H(\underline{\sigma}) < l$ has been found, one knows that the answer to the question above is "yes", and the algorithm can stop.

In Fig. 1.4, the probability p that a tour of length smaller than l exists, is shown [9] as a function of the rescaled length $\Phi = l/\sqrt{nA}$. One observes that there is a strong

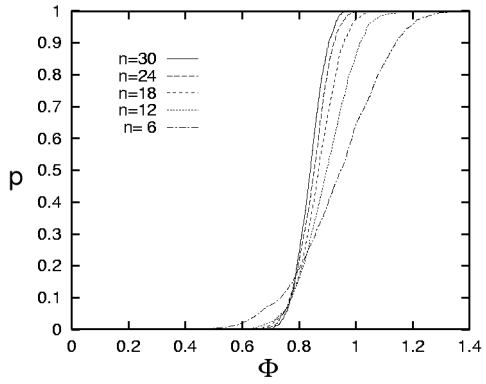

Figure 1.4: Probability p that the minimum tour of n cities in plane of area A is shorter than $\Phi = l/\sqrt{nA}$, for different number n of cities. Reprinted from Ref. [9] with permission from Elsevier.

increase in p when increasing l and that the curves for different sizes cross close to $\Phi = 0.78$. Hence, there seems to be a fundamental change in the nature of the question close to $\Phi = 0.78$, which resembles a *phase transition* in physical systems, see Sec. 5.1.

One can show [9] that all probability curves for different sizes n fall on top of each other, when p is plotted as a function of $r = (l/\sqrt{nA} - 0.78)n^{2/3}$. In this sense, the problem exhibits a certain type of universality, independent of the actual number of cities.

In Fig. 1.5 the running time of the TSP algorithm, measured by the number of times a position of some city is assigned, when creating permutations, is shown as a function of the parameter r. One observes that, close to the point, where exactly half of the instances have a shorter (minimum tour) length than l, the running time is maximal. This means that close to the phase transition, the problem is hardest to solve. It is a well known phenomenon in physics that, for the computer simulations of phase diagrams, the running time is maximal close to phase transitions. It is one of the main purposes of this book, to understand how this coincidence arises for combinatorial optimization problems.

Finally, we mention that it is easy to understand why the running time is small for large as well as for small values of Φ (i. e., l):

- For very small values of l, even the two closest cities have a larger distance than the given value of l, hence the algorithm can stop immediately after one step.
- For very large values of l, even the first permutation of the cities has a total distance smaller than l, i. e., the algorithm stops after $n - 1$ steps.

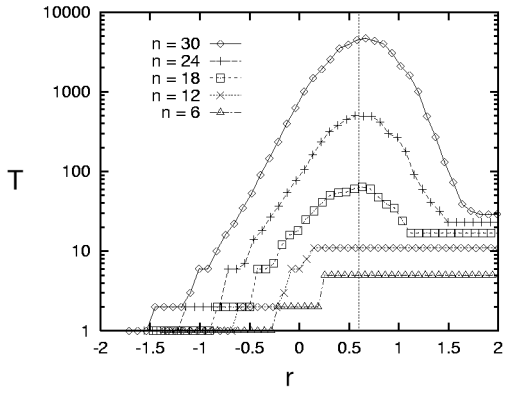

Figure 1.5: Running time for branch-and-bound algorithm as a function of $r = (l/\sqrt{nA} - 0,78)n^{2/3}$, for different number n of cities. The vertical line indicates where half of the instances have a shorter tour length than l. Reprinted from Ref. [9] with permission from Elsevier.

□

A better understanding of problems and algorithms may help to approach the solution of the ultimate task: to find fast algorithms such that larger instances can be handled efficiently. This is the main part of the work of computer scientists working in algorithmic design. Also in this field, physics has recently proved its strength. For example the *simulated annealing* technique originates in physics, see Chap. 2. The basic idea is to mimic the experimental cooling of a system to find the low-temperature, i. e., low-energy, behavior. This technique has the advantage of being widely applicable in very different fields. The disadvantage is that it does not guarantee finding the global optimum. For many problems in practice, it is, however, sufficient to find some local optima, which are close to the global one. Much more recently, the deep insight into the structural properties of the solutions of several hard optimization problems gained by statistical-mechanics analysis, has led to the proposal of a new class of statistical inference algorithms, called *survey propagation*. These algorithms are discussed in detail in Chaps 9 and 10.

1.3 Textbooks

In the following list, some useful basic textbooks are suggested.

- T. H. Cormen, S. Clifford, C. E. Leiserson, R. L. Rivest: *Introduction to Algorithms*, MIT Press, 2001
  This book is considered as the standard introduction to algorithms and data structures and provides a good foundation in the field which proves to be useful for theoretical studies and especially for practical applications and implementations of algorithms at all levels.

- A. V. Aho, J. E. Hopcroft, J. D. Ullman: *The design and analysis of computer algorithms*, Addison-Wesley, 1974
  This book merges the field of practical applications of algorithms and theoretical studies in this field.

- M. R. Garey and D. S. Johnson: *Computers and intractability*, Freeman, New York, 1979
  This book is a traditional standard text book on complexity theory. It concentrates on the part of theoretical computer science related to hard problems. The basic classes of problems are defined, many fundamental problems are explained, and their relationship is proved. In addition, the book contains a huge list of combinatorial problems which may serve as a source of inspiration for further research.

- C. H. Papadimitriou and K. Steiglitz: *Combinatorial Optimization*, Prentice-Hall, 1982
  This book gives a good introduction to the field of combinatorial optimization. All relevant basic problems and algorithms are explained. It exists in an economic paperback edition.

- A. K. Hartmann and H. Rieger: *Optimization Algorithms in Physics*, Wiley-VCH, Berlin, 2001
  This text book shows how optimization algorithms can be applied to many problems in physics. It explains the transformations needed to convert physical problems into optimization problems, and presents the algorithms needed to solve these problems.

- M. Mézard, G. Parisi, and M. A. Virasoro: *Spin glass theory and beyond*, World Scientific, Singapore, 1987.
  This book gives an introduction to the statistical-mechanics theory of spin glasses, together with reprints of the most important papers. It discusses also first applications of spin-glass methods to non-physical problems, including neural networks and combinatorial optimization.

- K. H. Fisher and J. A. Hertz: *Spin Glasses*, Cambridge University Press, 1991
  This book introduces the methods of statistical physics needed to study phase transitions in optimization problems. In this text, all techniques are applied to spin glasses, for which the methods were originally developed.

Bibliography

[1] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, Science **220**, 671 (1983).

[2] M. Mézard and G. Parisi, J. Physique **47**, 1285 (1986).

[3] N. J. Cerf, J. Boutet de Monvel, O. Bohigas, O. C. Martin, and A. G. Percus, Journal de Physique I **7**, 117 (1997).

[4] A. G. Percus and O. C. Martin, J. Stat. Phys. **94**, 739 (1999).

[5] A. Chakraborti and B. K. Chakrabarti, Eur. Phys. J. B **16**, 677 (2000).

[6] D. S. Dean, D. Lancaster, and S. N. Majumdar, arXiv preprint `cond-mat/0411111`.

[7] M. Mézard, G. Parisi, and M. A. Virasoro, *Spin Glass Theory and Beyond* (World Scientific, Singapore, 1987).

[8] K. H. Fisher and J. A. Hertz, *Spin Glasses* (Cambridge University Press, 1991).

[9] I. P. Gent and T. Walsh, Artif. Intell. **88**, 349 (1996).