About Traveling Salesmen and Telephone Networks – Combinatorial Optimization Problems at High School

Andreas Schuster, Würzburg

Abstract: This article introduces an investigation dealing with the question of what role the mathematical discipline ”combinatorial optimization” can play in mathematics and computer science education at high school. Combinatorial optimization is a lively field of applied mathematics and computer science that has developed very fast through the last decades.

Kurzreferat: Der Artikel stellt eine Untersuchung vor, die sich mit der Frage beschäftigt, welche Rolle das mathematische Teilgebiet der Kombinatorischen Optimierung, einer Disziplin, die im Umfeld von angewandter Mathematik und Informatik in den letzten Dekaden eine stürmische Entwicklung durchlaufen hat, für den Mathematik- und Informatikunterricht am Gymnasium spielen kann.

ZDM-Classification: N70, N90, U70

1 Combinatorial Optimization

Combinatorial optimization is a lively discipline of modern applied mathematics and computer science, but it cannot be said that it is a normal topic of today's mathematics or computer science education at German high school. Research is done at the moment dealing with the question, whether it is possible to bring certain topics from that field of mathematics and computer science into school (e. g. Lutz-Westphal, 2003; Schuster, 2001a,b, 2002, 2004). This article will show some research of the author together with some of his results.
A popular introduction to combinatorial optimization can be found in Gritzmann and Brandenberg, 2002; an up-to-date-textbook about that discipline is Cook et al., 1998.

1.1 Typical Problems in the Field of Combinatorial Optimization

First of all two problems in the field of combinatorial optimization are introduced that were subject to projects with high school students.

1.1.1 The Traveling Salesman Problem (TSP)

Which roundtrip can a salesman (see figure 1), who lives at village 1 and wants to visit his customers at the other villages, choose if the following conditions should be fulfilled:

- He wants to reach all the other villages exactly once.

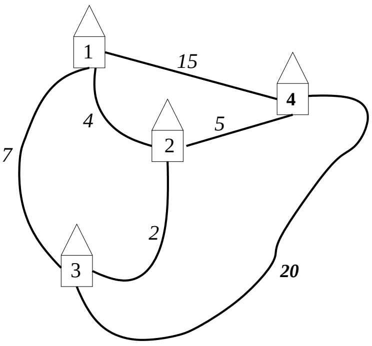

Figure 1: TSP/MST

- He doesn't want to reach village 1 during his journey again before returning at the end of the trip.
- At the end he returns to village 1.
- The ”costs” of his journey should be as low as possible.

The label of a road indicates the ”costs” of that way in a certain unit (e. g. time for traveling).

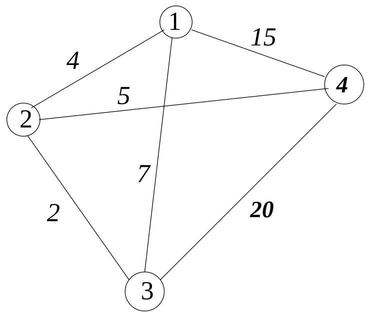

Figure 2: Graph for TSP/MST

Modelling (see figure 2) that problem using graph theory one gets the following problem (*traveling salesman problem, optimization version; TSPO*):
Given a complete graph that has at least three nodes and whose edges are labelled (without loss of generality) with positive integers.
A ”roundtrip” has to be found that visits every node exactly once (Hamiltonian circuit) and that causes minimal costs.

In complexity theory it is customary to use decision problems instead of optimization problems. The associated decision problem is (*traveling salesman problem, decision version; TSPD*):
Given a positive integer B and a complete graph that has at least three nodes and whose edges are labelled (without loss of generality) with positive integers.
Is there a ”roundtrip” that visits every node exactly once (Hamiltonian circuit) and whose costs are less or equal B?

Because the costs of a Hamiltonian circuit are an efficiently computable sum function it is easy to see that an efficient algorithm solving TSPO will lead to an efficient algorithm solving TSPD. By binary search one can see that the converse implication is also true.

1.1.2 The Minimum Spanning Tree Problem (MST)

Using the same pictures as for the traveling salesman problem (figure 1 and 2) the minimum spanning tree problem can be formulated as follows:
Some villages shall be connected by a telephone network. In which way have the villages to be connected in figure 1 along the roads so that everyone can make telephone calls to every person and that the network is as "cheap" as possible? The label of a road represents the "costs" of that part of the cabling in a certain unit.

Transforming that problem to graph theory you get the *optimization version of the minimum spanning tree problem (MSTO)*:
Given a connected graph with at least two nodes whose edges are labelled (without loss of generality) with positive integers.
A subgraph has to be found which is connected, has no circuit, includes all nodes and causes minimal costs (minimum spanning tree).

The corresponding *decision problem (MSTD)* is:
Given a positive integer B and a connected graph with at least two nodes whose edges are labelled (without loss of generality) with positive integers.
Is there a connected subgraph without any circuit that includes all nodes (spanning tree) and whose costs are less or equal B?

By the same argument used for TSPO/TSPD you can see, that an efficient solution for MSTO implies an efficient solution for MSTD and vice versa.

1.2 The Complexity Classes P and NP

One observation you can make is that combinatorial optimization problems can often be solved in an obvious way by *exhaustive search* (a brute force algorithm): Construct every possible solution (that's "in principle" possible, because there are only a finite number of possible solutions; but see Soare, 1987 for problems in the context of that "principle possibility") and then choose an optimum. This trivial algorithm isn't acceptable in general because of its time properties (an example will follow). This section introduces *in an informal way* certain concepts of complexity theory that are important to describe time properties of problems in the field of combinatorial optimization. A precise and formal introduction can be found in Garey and Johnson, 1997.
The complexity class **P** consists of all decision problems for which a solution algorithm exists that needs only time bounded from above by a polynomial in the input size (*"efficient algorithm"*). MSTD belongs to that class. One can see that this is correct by the algorithm of Prim (see algorithm 1) for MSTO and the remark made above about the connection between MSTD and MSTO. The class **NP**, roughly speaking, consists of all decision

Algorithm 1 Algorithm of Prim – Generic Version

Require: $(E, \mathcal{K}, c)$ is a connected graph with $|E| \geq 2$ whose edges are labelled with positive integers $c(k)$, $k \in \mathcal{K}$; $s \in E$ arbitrary

$\mathcal{S} := \{s\}$;
$\mathcal{L} := \emptyset$;
while $S \neq E$ **do**
  Choose an edge $k \in \mathcal{K} \setminus \mathcal{L}$ with minimum $c(k)$ so that one endpoint of k is in S and the other endpoint e of k is not in S;
  $S := S \cup \{e\}$;
  $\mathcal{L} := \mathcal{L} \cup \{k\}$;
end while

Ensure: $(E, \mathcal{L})$ is a minimum spanning tree of the graph, i. e. a solution of MSTO

problems with the following property: For all instances for which the answer to the decision question is "yes" there exists a "certificate" that the answer "yes" is correct, whose length is bounded from above by a polynomial in the problem size and from which the correctness of this answer can be derived by an efficient algorithm. Obviously TSPD is an element of **NP**. A lot of theoretical and practical problems also belong to **NP**.
As you easily can see, the following property is correct: $\mathbf{P} \subseteq \mathbf{NP}$. Whether the equality sign holds or not, isn't known so far (*P-NP-problem*); it is an important open question of complexity theory. In **NP** there are problems that are "most difficult" (*NP-complete problems*) in the following sense: If an efficient algorithm will be found some day for any of these problems, then there exist efficient algorithms for all problems in **NP**, that means "$\mathbf{P} = \mathbf{NP}$" would be correct. Researchers today believe (*P-NP-conjecture*) that $\mathbf{P} \neq \mathbf{NP}$; therefore efficient algorithms for **NP**-complete problems don't exist. TSPD is such a **NP**-complete problem. If the **P-NP**-conjecture is right, then there is no efficient algorithm for TSPD and therefore also none for TSPO (see the remark above about the connection between TSPD and TSPO). That means that every algorithm which solves all instances of TSPO will show the same time properties as the obvious exhaustive search algorithm (*exponential time algorithm*). Let's estimate the time solving TSPO by exhaustive search: In a complete graph with $n \geq 3$ nodes there are

$$\frac{(n-1)!}{2}$$

Hamiltonian circuits. If we assume that a computer needs at least $10^{-9}s$ to deal with any of these circuits,

then the algorithm will need for $n = 28$ nodes at least $172 \cdot 10^9$ years to finish its work. That is much more than the estimated age of the universe! The exponential growth in the number of the possibilities which is the reason for this time growth is called *combinatorial explosion*.
One of the consequences of this discussion is that students dealing with this kind of problem explore an actual frontier of research and that they learn two important dimensions of the concept of an algorithm: They see that time is an important category in evaluating algorithms and they learn from experience that an obvious and natural solving strategy ("brute force") will lead to inacceptable procedures because of time properties. This experience supplements the classical picture of mathematics with an up-to-date facet, which is important in modern applications of mathematics and computer science.

2 Combinatorial Optimization at High School

2.1 Properties of Combinatorial Optimization as a Subject of Education

In the following you can find certain principles to justify combinatorial optimization as a subject of mathematics and computer science education at high school. These principles are only enumerated; for a discussion see Schuster, 2004.

1. Dealing with combinatorial optimization not only contributes to developing mathematical systems in mathematics and computer science education but also forces students of different ages to explore real world problems developing and using techniques from mathematics and computer science.

2. These topics emphasize mathematical concepts (e. g. mathematics of modern applications) and aspects of mathematical working (e. g. constructing mathematical models) that aren't given special emphasis in a traditional mathematics curriculum. In this aspect they complement traditional mathematics education.

3. Students can rediscover mathematical ideas that also help solving problems that exist in reality.

4. Combinatorial optimization offers plenty of material for developing heuristic strategies.

5. Modern tools, especially the computer, can be used in a very meaningful manner: as an aid and as a subject of discussion (e. g. fundamental limits of an algorithmic machine; see the remarks above about the **P**-**NP**-conjecture and its importance to problems like TSPO).

6. Students get to know modern fields of work of mathematics and computer science and can explore frontiers of today's research.

7. Problems in the field of combinatorial optimizations have connections with other fields of science and research (operations research, economy etc.) and with traditional topics of mathematics education (e. g. calculus, analytical geometry etc.). Therefore they can be integrated into the mathematics curriculum in a natural way enriching today's mathematics education.

8. The natural use of "new" methods of teaching and learning (e. g. learning by performing projects) in the context of combinatorial optimization problems can serve as a preparation to the learning and working structures of the world of adults.

2.2 Survey of Empirical Studies and Theoretical Investigation

2.2.1 Presentation of the Projects

Five projects dealing with combinatorial optimization were done during the last three years serving as an empirical foundation for the research of the author:

- Project I: TSP in a 9^{th} grade of a German high school (February until May 2000); MST in the same 9^{th} grade (May until June 2000)

- Project II: TSP and MST at a workshop at the TU Munich for 9^{th}- and 10^{th}-graders from high school (November 2000)

- Project III (resumption of project I; the students were to a great extent the students of project I): MST in a 10^{th} grade of a German high school (February until July 2001) including a short introduction to the SPP (the *shortest path problem, SPP*, is another problem in the field of combinatorial optimization not mentioned above. The details about that problem can be found in Cook et al., 1998, pages 19 – 36)

- Project IV and V: SPP with 11^{th}- and 12^{th}-graders at workshops at the University of Würzburg in July 2001 and July 2002

These studies and the theoretical considerations of the author should serve as a "prelude" to the development of a curriculum. The activities of the students were observed to determine, what they were doing, when they worked autonomously on combinatorial optimization problems. The problems were formulated in a very open manner. Partially the students dealt with the problems simulating working in a software development company. They formed small teams; sometimes these teams dealt

with the same part of a problem, sometimes with different parts. In a ”plenary assembly” all students and the teacher discussed the results of each single team. That means that teamwork and dicussion alternated.
The projects were recorded on video tape, which will be interpreted using the methods of interpretative teaching evaluation (Maier and Voigt, 1994, 1991). Furthermore the students had to make written documentations about central parts of their considerations.

2.2.2 Aspects of Investigation

The theoretical concepts which are developed at the moment and the investigation of the material deal with the following questions:

1. *Didactical phenomenology:* What are the main properties of a didactical phenomenology (see Freudenthal, 1983) of mathematical structures of combinatorial optimization?
2. *Simplification:* Is it possible to simplify subject matters and methods of proof of combinatorial optimization in such a way that they can be taught at high school, and in which way is that possible, if the answer of the former question is ”yes”?
3. *Design of teaching:* Which methods are especially suitable for teaching these topics?
4. *Problem of heuristics:* In what way and to what extent can topics in the field of combinatorial optimization be used to develop heuristic strategies?
5. *Problem of approximation strategy:* Which approximative solutions do students develop when they face the problem that exact solutions can't be found in acceptable time?
6. *Learning of concepts:* What do topics in the field of combinatorial optimization contribute to learning of concepts in mathematics and computer science education?
7. *Context problem:* What relationship exists between combinatorial optimization and the traditional main ideas of mathematics education and education in computer science?

2.2.3 Example from the Transcripts: Time Complexity of an Algorithm and the Concept of a Solution

As an example I will use the following scene from the transcripts accompanied by a short interpretation. The students had to search for a solution of a TSPO in the form of a drilling problem for circuit boards, simulating developer teams of a software company. At the end of that phase one of the 9^{th}-graders (”S”) explains the results of his team to the plenary assembly of all students (translation of the author):

> S: Yes, it was suggested that all possibilities should be tested, and the computer should always add up the sums, i.e., it always adds the distances and then takes the shortest one. However if you do so in practice with thousand holes [that must be drilled on the circuit board; annotation of the author] and so, and you pass through all possibilities, there exist ca. one million possibilities. Until you will finish, you need hundreds of years and no company has actually so much time, that it can wait three hundred years, until the job will be done. Well – we have now reflected: If we could do now the same, that it [the computer or the algorithm; annotation of the author] tests all possibilites, but with a restriction, that it, when it has a point, that it doesn't connect it with all points, but only with those points, which have a certain distance, i.e. a certain maximum distance. Thus a lot of possibilities are cancelled, and there are still only a few hundred or a few thousand possibilites and of course these are quickly tested.

The teams that reported their results before the student used exhaustive search to solve the problem. They didn't realize the importance of time in any way. S rejects that solution, he notices that time is an important dimension of the problem. At that age growth properties of functions are hardly known. Therefore it is remarkable that he notices the growth in the number of the possibilities as a significant phenomenon. The way he speaks gives the impression that the time argument is quite natural and obvious for him. Later in the scene (that part is not quoted here) one can assume that the errors in the numerical calculation of S are partly explained by his misconception that the number of possibilites increases by n^3, if n is the number of holes or nodes, accompanied by problems using numerals. Therefore S believes implicitly that there is a polynomial growth; he doesn't discover the phenomenon of combinatorial explosion. He concludes from his (partly wrong) considerations that he needs another *kind of solution*, i. e., *his concept of solution changes.* His (approximative) strategy tends towards a *decomposition approach*. *Local optimality* becomes a fundamental concept for him. He uses a *divide-and-conquer-strategy.*

2.2.4 Some Results

The author distinguishes *four kinds of activities* in the context of problems in the field of combinatorial optimization:

I. Mathematical modelling of a real world problem

II. Searching for and discovering/constructing algorithms

III. Investigation of algorithms

IV. Proving the correctness of an algorithm

For a more detailed discussion of these activities see Schuster, 2002 and 2004. The following results deal with algorithms in the field of combinatorial optimization in education, not with algorithms as educational subjects in general!

- 9^{th}- and 10^{th}-graders are able to perform activites of category I and II to a great extent autonomously. They are very motivated and creative and are able to find standard techniques known in literature. Local optimality and greedy strategies are preferred heuristics.
- 9^{th}- and 10^{th}-graders can develop activities of category III only partly. For example they are able to show that a suggested algorithm isn't correct, but they can't develop a correct algorithm autonomously to a level of precision that is customary in mathematics or computer science.
- Activities of category IV are almost completely beyond the scope of 9^{th}- and 10^{th}-graders, but towards the end of that period this seems to change.
- 11^{th}-graders and 12^{th}-graders are able to perform activities of category III and IV autonomously.

References

William J. Cook, William H. Cunningham, William R. Pulleyblank, and Alexander Schrijver. *Combinatorial Optimization*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, Inc., New York, Chichester, Weinheim, Brisbane, Singapore, Toronto, zweite edition, 1998.

Hans Freudenthal. *Didactical Phenomenology of Mathematical Structures*. D. Reidel Publishing Company, Dordrecht, Boston, Lancaster, 1983.

Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman and Company, New York, neunzehnte edition, 1997.

Peter Gritzmann and René Brandenberg. *Das Geheimnis des kürzesten Weges. Ein mathematisches Abenteuer*. Springer-Verlag, Berlin, Heidelberg, New York, erste edition, 2002.

Brigitte Lutz-Westphal. http://www.math.tu-berlin.de/~westphal/projekt, 2003.

H. Maier and J. Voigt, editors. *Interpretative Unterrichtsforschung*, volume 17 of *Untersuchungen zum Mathematikunterricht*. Aulis Verlag Deubner & CO KG, Köln, 1991.

H. Maier and J. Voigt, editors. *Verstehen und Verständigung. Arbeiten zur interpretativen Unterrichtsforschung*, volume 19 of *Untersuchungen zum Mathematikunterricht*. Aulis Verlag Deubner & CO KG, Köln, 1994.

Zbigniew Michalewicz and David B. Fogel. *How to Solve It: Modern Heuristics*. Springer-Verlag, Berlin, Heidelberg, New York, zweite edition, 2000.

Andreas Schuster. Diskrete Optimierungsprobleme als Unterrichtsgegenstand im Umfeld von Mathematik und Informatik. In Gabriele Kaiser, editor, *Beiträge zum Mathematikunterricht 2001 – Vorträge auf der 35. Jahrestagung für Didaktik der Mathematik vom 5. bis 9. März 2001 in Ludwigsburg*, pages 552 – 555, Hildesheim, Berlin, 2001a. Gesellschaft für Didaktik der Mathematik (GDM), Verlag Franzbecker.

Andreas Schuster. Zeitkomplexität und Approximation als Unterrichtsgegenstand in der neunten Klasse des Gymnasiums. Auszug aus einer Fallstudie. *mathematica didactica*, 1(24):31 – 50, 2001b.

Andreas Schuster. Von Handlungsreisenden und Telefonnetzen: Software bei der Behandlung von Optimierungsproblemen. In Wilfried Herget, Rolf Sommer, Hans-Georg Weigand, and Thomas Weth, editors, *Medien verbreiten Mathematik. Bericht über die 19. Arbeitstagung des Arbeitskreises „Mathematikunterricht und Informatik“ in der Gesellschaft für Didaktik der Mathematik e. V. vom 28. bis 30. September 2001 in Dillingen*, pages 153 – 163, Hildesheim, Berlin, 2002. Gesellschaft für Didaktik der Mathematik (GDM)/Arbeitskreis Mathematikunterricht und Informatik, Verlag Franzbecker.

Andreas Schuster. Kombinatorische Optimierung als Gegenstand der Gymnasialdidaktik im Umfeld von Mathematik- und Informatikunterricht. Habilitationsschrift. In Vorbereitung, 2004.

Robert I. Soare. *Recursively Enumerable Sets and Degrees. A Study of Computabel Functions and Computably Generated Sets*. Perspectives in Mathematical Logic. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, 1987.

Author

Schuster, Andreas, Lehrstuhl für Didaktik der Mathematik, Bayerische Julius-Maximilians-Universität Würzburg, Am Hubland, 97074 Würzburg
E-Mail: schuster@mathematik.uni-wuerzburg.de