Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

MARTIN FRANK[1]
ARMIN FÜGENSCHUH[2]
MICHAEL HERTY[3]
LARS SCHEWE[4]

The Coolest Path Problem[5]

[1]RWTH Aachen University, Mathematik, Templergraben 55, D-52056 Aachen
[2]Zuse Institut Berlin, Takustr. 7, D-14195 Berlin
[3]RWTH Aachen University, Mathematik, Templergraben 55, D-52056 Aachen
[4]TU Darmstadt, Fachbereich Mathematik, Schlossgartenstrasse 7, D-64289 Darmstadt
[5]to appear in: Networks&Heterogeneous Media

ZIB-Report 09-37 (November 2009)

THE COOLEST PATH PROBLEM

MARTIN FRANK, ARMIN FÜGENSCHUH, MICHAEL HERTY, AND LARS SCHEWE

Abstract. We introduce the coolest path problem, which is a mixture of two well-known problems from distinct mathematical fields. One of them is the shortest path problem from combinatorial optimization. The other is the heat conduction problem from the field of partial differential equations. Together, they make up a control problem, where some geometrical object traverses a digraph in an optimal way, with constraints on intermediate or the final state. We discuss some properties of the problem and present numerical solution techniques. We demonstrate that the problem can be formulated as a linear mixed-integer program. Numerical solutions can thus be achieved within one hour for instances with up to 70 nodes in the graph.

Continuous and discrete optimization are at present two distinct areas of mathematics. From time to time, discrete optimizers stumble over a problem which has some intrinsic nonlinear continuous structure, sometimes modeled using partial differential equations. Then they most likely would try to get rid of these continuous parts, such that a pure combinatorial problem remains. Similarly, if a person with a background in continuous optimization gets involved with a problem that involves discrete decisions, he or she would most likely try to relax the discontinuities to some continuous constraints, in order to apply some well-understood methods of the field. For both of them it is true that *if one only owns a hammer then every problem must be a nail*. However it is also true that if one always stays within its own cosy corner of the world, nothing new can emerge from that.

Our research is motivated by the fact that both worlds can inspire the respective other by sharing ideas and methods. So to start the discussion at some point we combine two problems into a new one that was not studied before (to the best of our knowledge). From the discrete world we consider the shortest path problem on a directed graph. The contribution from the continuous world is the heat conduction problem. Both problems are combined into a new optimization problem, which we suggest to coin *the coolest path problem*.

1. The Coolest Path Problem

We consider the following problem. Given is a directed graph $D = (V, A)$ with vertex set V and arc set A, and two distinct nodes $v, w \in V$. An v-w-path P in D of length n is defined as a sequence of vertices and arcs of the form $P = (v_0, a_1, v_1, a_2, v_2, \ldots, v_{n-1}, a_n, v_n)$, where $v_0 = v, v_n = w, a_i = (v_{i-1}, v_i)$, and the arcs in P are pairwise different. Imagine that a geometric object $\Omega \subset \mathbb{R}^3$ traverses the network from v to w. The initial temperature of the object is given by $u_0 : \Omega \to \mathbb{R}_+$. Associated with each arc $a \in A$ is a temperature $T_a(x) \in \mathbb{R}_+$ for $x \in \partial\Omega$. On each arc a the boundary of the object $\partial\Omega$ is exposed to the prevalent temperature $T_a(x)$ for $x \in \partial\Omega$ for a certain, arc-dependent time, so that the object is heated up or cooled down. At the end, at vertex w, the temperature distribution within the object is given by the path-dependent function $u_P : \Omega \to \mathbb{R}_+$. The *coolest path problem* (CPP, for short) asks for an v-w-path P such that the average temperature $\bar{u}_P := \frac{1}{vol(\Omega)} \int_\Omega u_P \, dV$ of the object at w is minimal. The coolest path problem thus combines the combinatorial problem of finding a shortest v-w-path, where "shortest" refers to the amount of absorbed heat on the path, which is modelled by the heat equation.

As a real-world application of this problem one might think of a production line where some product (the object) has to pass certain manufacturing steps. These steps impose some heating or cooling to the material. After the end of one step there is a number of other succeeding steps that have to be carried out afterwards, until the product reaches the output. At the end, the product should be as cool as possible.

Besides this basic version of the problem, there are natural variations and extensions which we also consider in the sequel.

(1) Other objective functions. For example, one can take the temperature $u_P(x)$ at a certain point $x \in \Omega$ or the maximum temperature $\max\{u_P(x) : x \in \Omega\}$ as objective functions.
(2) Temperature gradients. The goal here is to find a path P such that the norm of gradient $\|\text{grad}(u)(\cdot)\|$ is minimal, either at a given point x, at the maximum within Ω, or in the average.
(3) Restrictions along the path. The above objective functions, together with a lower or upper bound can be taken as constraints. In this case we have to deal with a feasibility problem (i.e., finding a path with the given property), or together with any other of the objective functions from above, as an optimization problem with further constraints on the path.
(4) Control problems. The goal is to achieve a final state, such as a desired heat distribution at vertex w, and finding a path such that the actual heat distribution is closest possible to the prescribed one.

Another interesting variant is the coolest Hamiltonian cycle (CHC, for short) in a digraph. In the classical Hamiltonian cycle (HC) problem one is interested in a tour (or cycle) through all nodes that starts and end at the same node, and enters and leaves every node exactly ones. We remark that HC is NP-complete, see the monograph by Garey and Johnson [7]. In the "cool" version, one wants to end up with the coolest possible object (with respect to some objective functional). The CHC can also be combined with all variations from the above list. We will demonstrate that our methods are also able to solve CHC as a by-product.

The combination of shortest path with heat conduction gives rise to the question whether some of the combinatorial shortest path algorithms can be modified to solve this new problem. We will demonstrate in the sequel that this is only possible in a very special case. In the general case we are in the same situation as with the general shortest path problem with negative arc weights and negative cycles. Thus we cannot give a simple combinatorial algorithm for its solution. Instead we will formulate the problem as a mixed-integer linear program, which can be solved numerically within the general linear programming (simplex) based branch-and-cut framework (see Schrijver [14] or Nemhauser and Wolsey [11] for an introduction).

To formally state the coolest path (or coolest cycle) problem we introduce some more notations. Denote by F one of the objective functionals from above. Select two distinct nodes $v, w \in V$. Let $\mathcal{P}_{v,w}$ be the set of all paths from v to w in D. The time for traversing arc $a \in P$ is denoted by τ_a. If we assume that the object Ω starts at time $t_0 := 0$ then the end time is $t^* := \sum_{a \in P} \tau_a$. Function $u(x,t)$ describes the heat distribution in the object at location x and time t, depending on path P. To be more precise, $t \mapsto u(x,t)$ depends only on those arcs that were traversed before time t, for all $x \in \Omega$ and $t \in [0, t^*]$. Note that each point in time $t \in [0, t^*]$ can be mapped onto an arc $a(t) \in P$ which the object traverses at time t.

Using this notation the problem can be formally stated as follows:

$$\min_{u;P \in \mathcal{P}_{v,w}} \quad F(u(x,t^*)) \tag{1}$$

$$\text{such that} \quad \frac{\partial u}{\partial t}(x,t) = k \cdot \frac{\partial^2 u}{\partial x^2}(x,t), \quad \forall\, x \in \Omega, \forall\, t \in [0,t^*], \tag{2}$$

$$\frac{\partial}{\partial n} u(x,t) = h \cdot (T_{a(t)}(x) - u(x,t)), \quad \forall\, x \in \partial\Omega, \forall\, t \in [0,t^*], \tag{3}$$

$$u(x,0) = u_0(x), \quad \forall\, x \in \Omega. \tag{4}$$

2. Mathematical Background

Before actually solving the problem at hand we start with a survey of the shortest path problem and the heat equation. From this study we can also show in which directions our methods can be extended.

2.1. **Shortest Paths in Graphs.** One major ingredient is the classical shortest path problem on (directed or undirected) graphs (SPP, for short). An instance of the SPP is defined by a directed weighted graph $D = (V, A, c)$, where $c : A \to \mathbb{R}$ are arc weights, and two distinct nodes $v, w \in V$. The cost (or length) of an v-w-path P is hereby defined as the sum of weights of its arcs, i.e., $c(P) := \sum_{a \in P} c_a$. The problem asks for an v-w-path P of minimal length. In this spirit the coolest path problem can be seen as a combination of a pure combinatorial problem with an objective function that takes the amount of absorbed heat along the path into account.

The mathematical study of the combinatorial shortest path problem in graphs can be dated back to the 1950s (see Schrijver [15]). There exists several algorithms for its solution.

The key observation that leads to *efficient*, i.e., polynomial time algorithms, is the property that all subpaths of a shortest path are as well shortest paths. However, this property holds if and only if the graph does not contain a negative cycle. In this case we can use an algorithm due to Moore [10], Bellman [2], and Ford [6], which has a running time proportional to the number of vertices cubed. In the special case that all edge weights are non-negative one can use Dijkstra's algorithm [4] which has only a quadratic running time.

In the general shortest path case negative weights (and negative circles) are allowed. There is no efficient combinatorial algorithm known in that case, and it is most likely that no such algorithm exists (unless $P = NP$). A special case of the shortest path with negative cycles is the longest path problem, which asks for the longest possible path (also called critical path) between two distinct nodes. More general than this, one can consider the path problem with given length, where a path is sought which connects the two nodes with a path of a prescribed length (or to decide that no such path exists). This problem is also NP-hard. Later on, this problem will occur as a subproblem in one of our solution methods for the CPP.

The solution of the corresponding linear program from above is still integral, but most likely cycles (with negative sum of its arcs) will occur. In order to obtain cycle-free solutions, one can use the following model. Let a weighted digraph $D = (V, A, c)$ with arbitrary arc weights $c_a \in \mathbb{R}$ for all $a \in A$ be given. Select two distinct nodes $v, w \in V$. We define a set $A^* := A \times \{1, \ldots, |A|\}$ and introduce binary variables $z_{i,j,p} \in \{0, 1\}$ for all $(i, j, p) \in A^*$. If $z_{i,j,p} = 1$ then arc (i, j) is selected as the p-th arc in the v-w-path. Every arc of A can in principle occur in this v-w-path. Hence the number of elements in A, i.e., $|A|$, is an upper bound on the number of arcs in the path. Moreover we introduce binary variables $y_p \in \{0, 1\}$ for all $p \in \{1, \ldots, |A|\}$, where $y_p = 1$ indicates that the path consists of exactly $(|A| - p)$ arcs.

Using these definitions the shortest path problem with arbitrary arc weights can be formulated as follows:

$$\text{(5)}\min \quad \sum_{(i,j,p)\in A^*} c_{ij} \cdot z_{i,j,p},$$

$$\text{(6) s.t.} \quad \sum_{i:(i,w)\in A} z_{i,w,|A|} = 1,$$

$$\text{(7)} \quad \sum_{j:(v,j)\in A} z_{v,j,p} = y_p, \quad \forall\, p \in \{1, \ldots, |A|\},$$

$$\text{(8)} \quad \sum_{p\in\{1,\ldots,|A|\}} y_p = 1,$$

$$\text{(9)} \quad \sum_{p\in\{1,\ldots,|A|\}} z_{i,j,p} \leq 1, \quad \forall\, (i, j) \in A,$$

$$\text{(10)} \quad \sum_{i:(i,k)\in A} z_{i,k,p-1} = \sum_{j:(k,j)\in A} z_{k,j,p}, \quad \forall\, k \in V \backslash \{v, w\}, \forall\, p \in \{2, \ldots, |A|\},$$

$$\text{(11)} \quad y_p \in \{0, 1\}, z_{i,j,p} \in \{0, 1\}, \quad \forall\, (i, j) \in A, \forall\, p \in \{1, \ldots, |A|\}.$$

Constraint (6) forces the last arc of the path to end at node w. By constraints (6) the last arc of the path, which is the p-th arc within the path, connects node w. Exactly one arc is the first arc, which is modeled by (7) and (8). Constraints (9) ensure that every arc occurs at most once in the v-w-path. The connectivity of the paths is due to the flow conservation constraints (10).

We note, that it is possible to formulate this problem only using a set of variables that indicates whether an arc is in the path or not. In that case we can ensure the condition that no cycle occurs by adding suitable cut constratints to the model. This polyhedron has also recently been studied by Stephan [17]. However, for our later models we will need an explicit encoding of the order of the arcs in the path to help us compute the temperature distribution.

As a possible solution technique one can apply branch-and-bound or branch-and-cut, which re-introduces the integrality after its relaxation. This technique will be briefly described in the subsequent section.

This model can be replaced with a much simpler one if no negative cycles occur. We introduce variables

$$\text{(12)} \qquad z_{i,j} \in \{0, 1\}, \quad \forall\, (i, j) \in A.$$