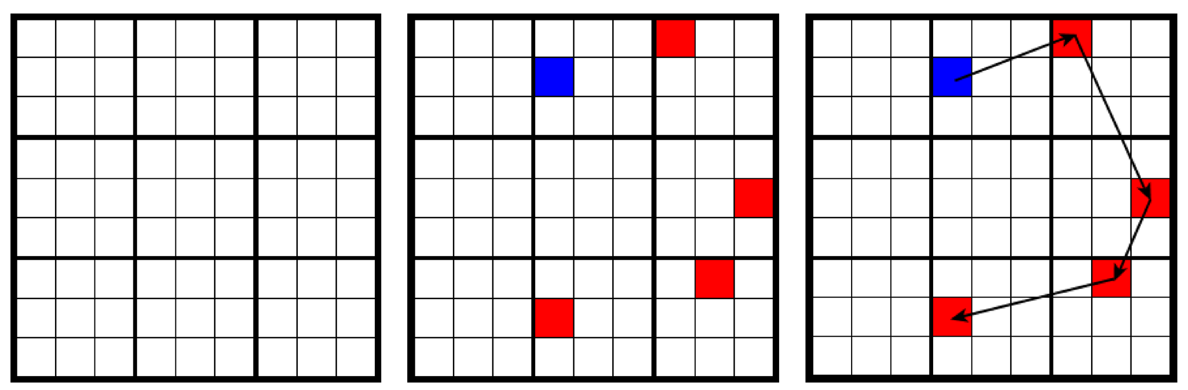
Heuristics for the Sudoku Distances Problem

Joel Miller, Oberlin College
Dr. Kim Factor[1], Mentor
1: Department of Mathematics, Statistics, and Computer Science, Marquette University

Introduction

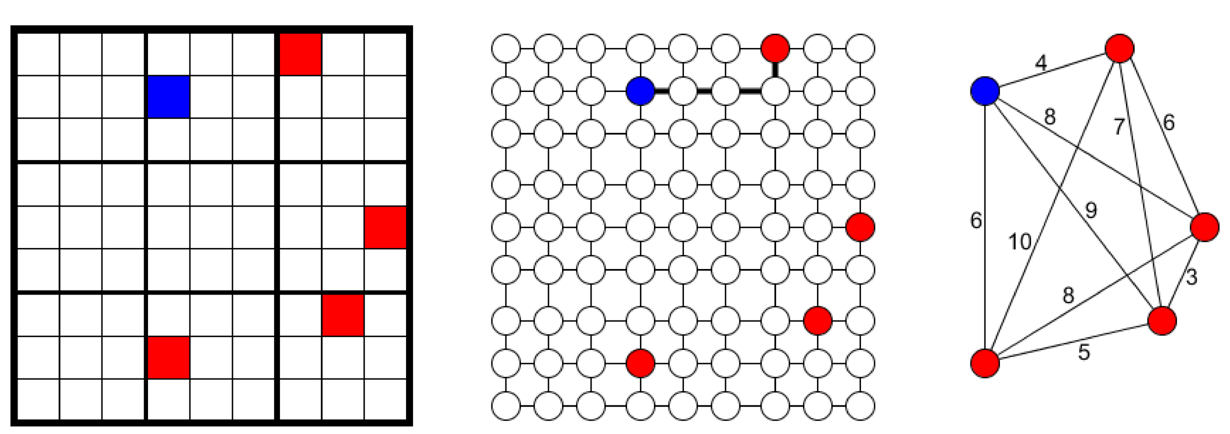
The Sudoku Distances Problem (or SDP) is a variation of the Traveling Salesman Problem (or TSP):



Given a start square s and a set of squares to visit Q on a Sudoku board, find a minimum-length path from s through all of Q.

Finding heuristics for the SDP can help us understand the more general TSP, an important NP-Complete problem.

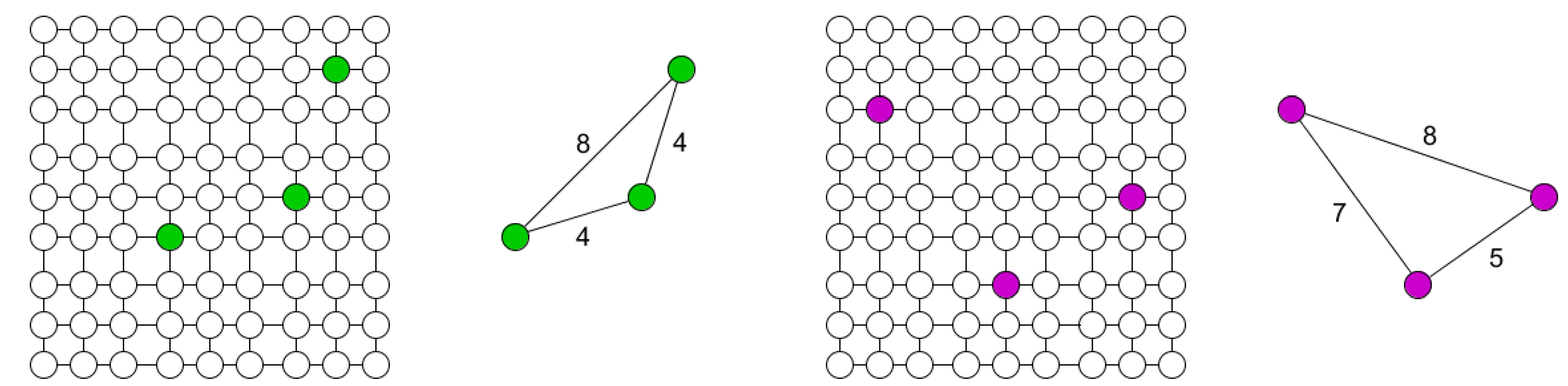
Definitions



B G_B G

- $B, G_B,$ and G are successively more abstract representations of the Sudoku Board.
- $d(a, b)$ denotes the graph distance between squares a and b in G_B, and signifies edge weights in G. **Example:** In G_B, s is 4 edges away from the uppermost square in Q.
- $\hat{d}(P)$ denotes the total distance of path P.
- P *crosses itself* if two edges in P cross in the geometric representation of P on B.
- $\hat{c}(P)$ denotes the number of times P crosses itself.
- A set of squares in B is *in-line* if an ordering of the squares by x-coordinate corresponds to an ordering by y-coordinate.

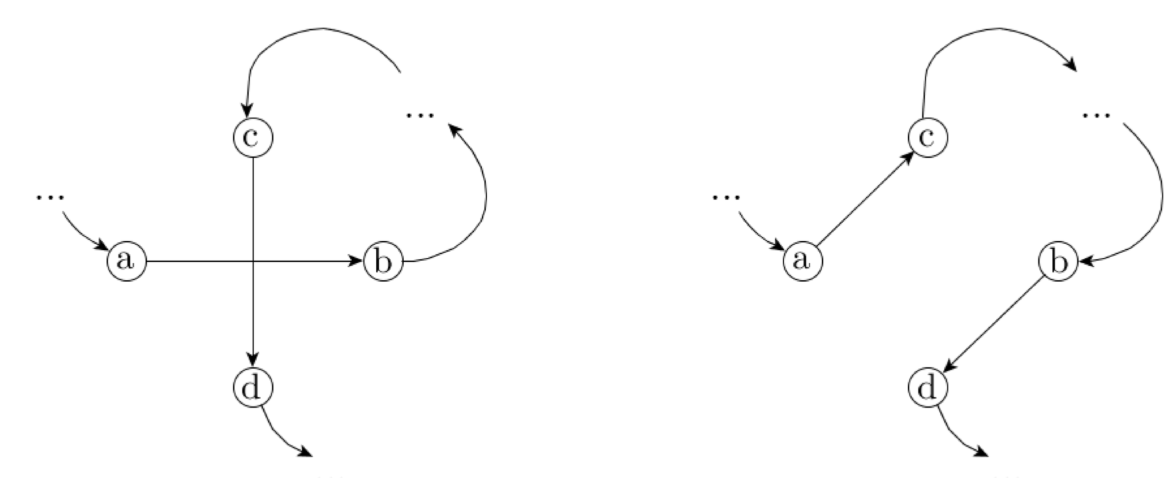
Methods



Many results stem from the observation that for in-line squares, $d(a, c) = d(a, b) + d(b, c)$. For squares that are not in-line, $d(a, c) > d(a, b) + d(b, c)$.

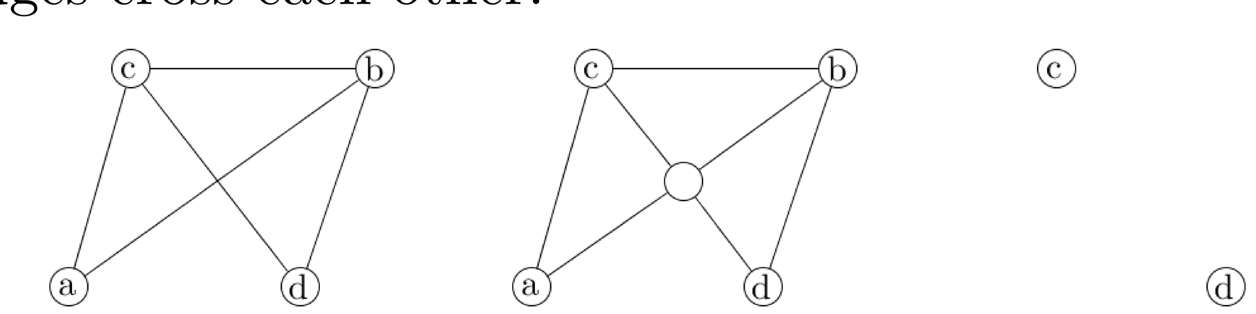
Theoretical Results

A procedure for un-doing crosses in paths: If $P = \dots ab \dots cd \dots$ and (a, b) crosses (c, d), $P' = \dots ac \dots bd \dots$ has no such cross.



P P'

Theorem 1. $\hat{d}(P) \geq \hat{d}(P')$.
Theorem 2. If P is optimal and $\hat{c}(P) = k$, repeated application of the *un-crossing procedure* produces P' such that $\hat{c}(P') = 0$. It follows that an optimal path never needs to cross itself.
We construct the *crossing graph* of G, G_C, which is a union of edge-specific subgraphs. G_C encodes information about which edges cross each other.



G G_C $G_C - G_{(a,b)}$

Theorem 3. $G_C - G_{(u,v)}$ is disconnected → every path that starts with (u, v) crosses itself.

Computational Results

Heuristic 1. Greedily construct P. At each step let $G' = G - \{v \in P \mid P \neq \dots v\}$. For all in-line trios a, b, c, set $G' = G' - \{b\}$ and set $d(a, c) = d(a, c)/2$.
Theorem 4. G_C can be constructed in $O(|E(G)|^2)$.
Heuristic 2. Greedily construct P. At each step construct G' and G'_C. For each edge e, if $G'_C - G'_e$ is disconnected, set $G' = G' - \{e\}$.

Scope

The formulation of the SDP suggests that each instance of the problem takes place on a 9-by-9 Sudoku board, which implies certain constraints. However, the theorems and heuristics shown here apply to any sized space and any configuration of points, so long as the same distance metric is used.

Future Work

- Prove the approximation ratio of each heuristic.
- Find heuristics that utilize the specific constraints of Sudoku.
- Prove the complexity class of the SDP. Although the SDP is closely related to the TSP, and the decision version of the problem is clearly in NP, there is no known reduction to prove NP-hardness.

Acknowledgements

The author would like to thank Dr. Kim Factor and Dr. Dennis Brylow for organizing Marquette's MSCS REU program. This research was made possible in part by a generous grant from the National Science Foundation, **NSF Award #ACI-1461264**.