Strahler Stream Order Inspired Gateway Shortest Path Subsets

F. Antonio Medrano and Richard L. Church

Project 301CR, GeoTrans Report 2013-12-01
December 2013

Photos courtesy of DOE/NREL

University of California, Santa Barbara
Department of Geography
1832 Ellison Hall
Santa Barbara, CA 93106
medrano@geog.ucsb.edu
church@geog.ucsb.edu

Strahler Stream Order Inspired Gateway Shortest Path Subsets

F. Antonio Medrano
Richard L. Church

Corridor Location Project
GeoTrans Laboratory
Department of Geography
University of California, Santa Barbara
December 31, 2013

This report has been developed as a part of the corridor location research project at the University of California, Santa Barbara. The goal of this project is to take a fresh look at the process of corridor location, and develop a set of algorithms that compute path alternatives using a foundation of solid geographical theory in order to offer designers better tools for developing quality alternatives that consider the entire spectrum of viable solutions. And just as importantly, as data sets become increasingly massive and present challenging computational elements, it is important that algorithms be efficient and able to take advantage of parallel computing resources. Please cite this report as: Medrano, FA, and RL Church (2013) "Strahler Stream Order Inspired Gateway Shortest Path Subsets" (Report #12-13-01), GeoTrans Laboratory, UCSB, Santa Barbara CA.

I. Introduction

The gateway shortest path problem (Church *et al.* 1992, Lombard and Church 1993) has been shown to be an effective method for efficiently generating sets of alternative routes on a raster network. Essentially a form of the constrained shortest path problem, a gateway path is the shortest path from an origin to a destination, constrained to also traverse through one or more specific intermediate points. While the gateway approach has shown great promise in being able to generate good paths with relatively little computational effort, thus far all techniques that have been developed to screen or review alternatives generated by the gateway model are manual approaches. Even the methodology employed by ESRI in their cost-distance model requires the user to manually review possible alternatives. What is needed is an approach that is capable of identifying good candidate gateway points given the set of gateway solutions. This candidate set would then comprise of points that are potentially "superior" in the sense that they lead to efficient, but *spatially* different solutions. In this report we explore a promising approach to identify such points based upon a process that was inspired by a technique in hydrology that is used to assign order to branches in a stream network.

The single gateway shortest path approach begins by the construction of two shortest path trees, one which is rooted at the origin and one which is rooted at the destination. A shortest path tree represents the shortest paths to all other nodes from a root or starting node. A shortest path tree can be easily generated by an algorithm such as Dijkstra's by starting at a given node and stopping when all nodes have been permanently labeled by a distance from the root node. By keeping track of the precedence nodes in the path, all arcs in the tree can be retrieved and a tree constructed after the algorithm has finished. The essence of the gateway process is to generate two trees and then use the two trees to identify the route and cost for a path that travels to a gateway node (along the tree rooted at the origin) and then on to the destination (along the tree rooted at the destination). This pathway is the least cost pathway that travels from the origin to the destination and is forced to travel via the gateway point. All single gateway shortest paths can be retrieved from these two shortest path trees. In addition, it is easy to compute the distances of all gateway shortest paths by adding the distances of the two distance labels (one for each of the two trees) at each node. From this it is easy to compute a cost surface that shows the cost of travel from the origin to the destination through each possible gateway node. This is computed as a feature of the cost distance function provided in ESRI's ArcMap. An example from ArcMap is given in the following pages. Figure 1 shows a cost grid that is used to compute the shortest routes. The output of the ESRI functionality is given as two rasters: one in which the optimal route is depicted and one of the composite cost surface. Figure 2 depicts a raster indicating the source node and the destination node, as well as the optimal route in green. In this case, the origin is in the southwest corner and the destination is the node in the northeast corner. The shades of pink in the background are associated with the cost distance from the origin. Figure 3 shows the composite gateway cost surface, comprised of the sum of the cost distance from the origin and the cost distance from the destination. The ESRI functionality is designed to produce the cost surface and the shortest route, but does not provide an easy way to peruse spatially different alignment alternatives other than to view the composite costs.

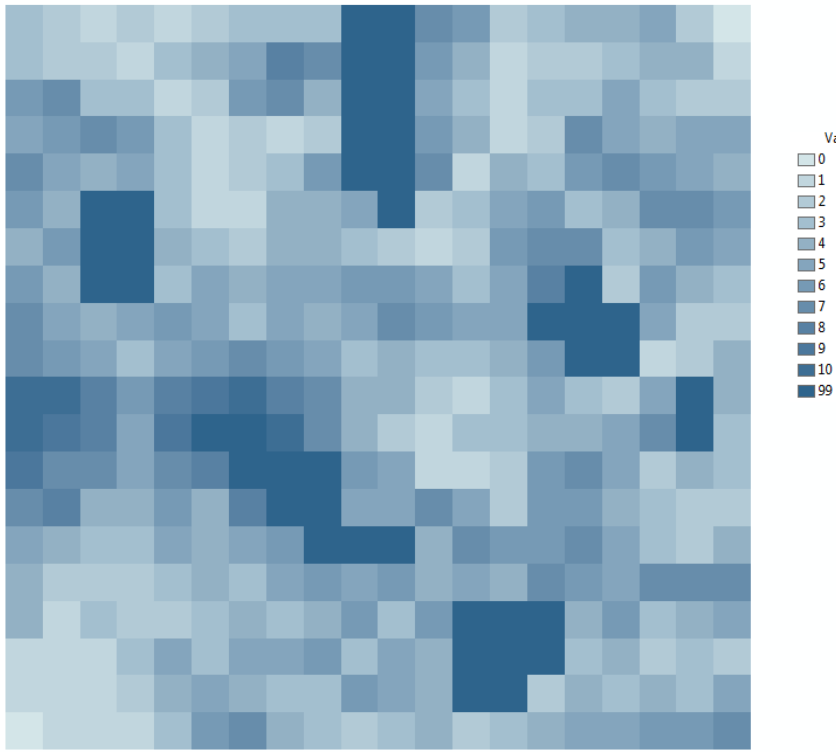


Figure 1. ArcMap 20x20 cost grid

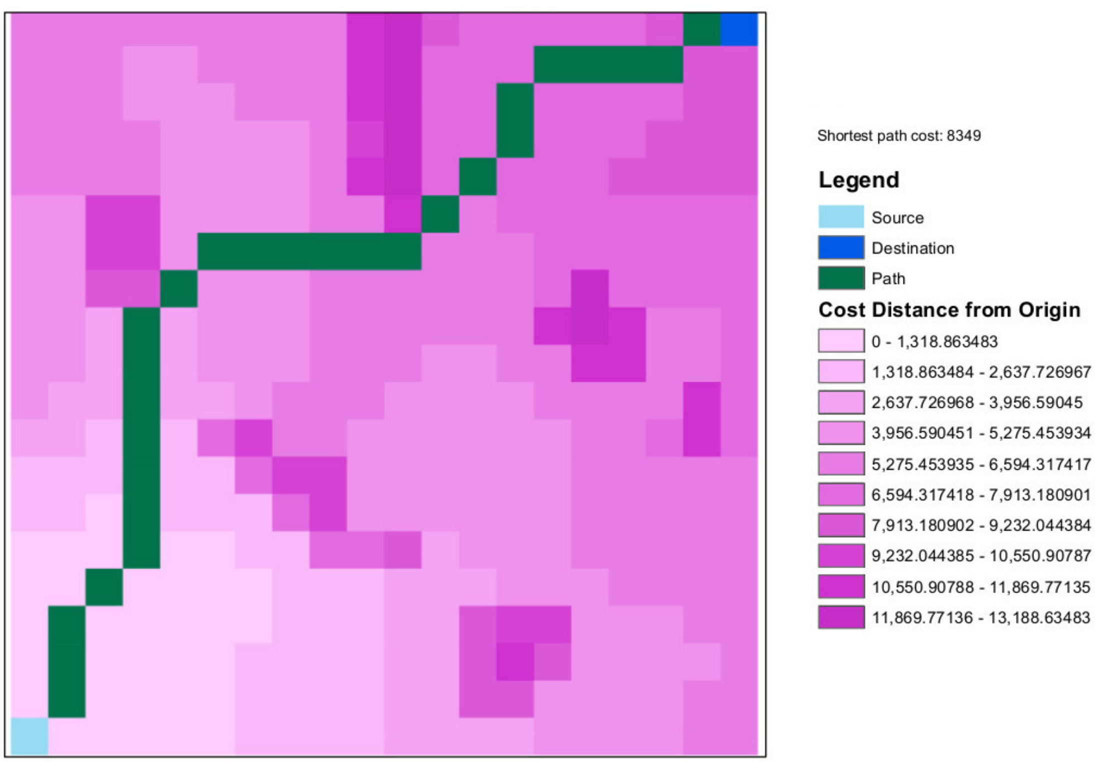


Figure 2. ArcMap 20x20 shortest path and cost distance from origin

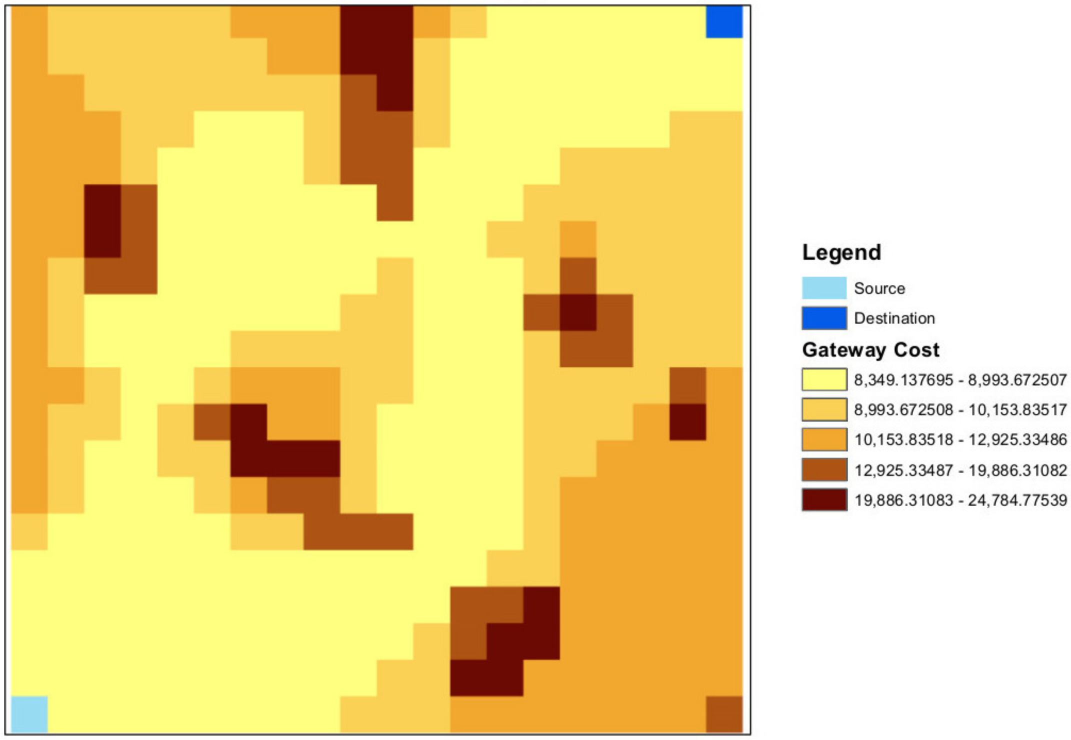


Figure 3. ArcMap 20x20 composite gateway cost surface

Church *et al.* (1992) developed an interface that allowed a user to retrieve and view any gateway path, as well as easily identify specific paths that were on the most efficient paths that were spatially different (using a Tchebychev function). In either case, the number of possible gateway path alignments to explore can be easily overwhelming. For example, a simple 80 row and 80 column raster contains 6400 cells all of which are possible gateway cells. When a more meaningful sized raster is employed in transmission routing (e.g. 1000 rows and 1000 columns), the number of gateway locations could be on the order of a million or more. As shortest path trees are "hydrologic" in structure, we have explored the use of Strahler Stream Order values as a method of identifying the principal limbs that represent potentially spatially different alternatives and sift through a potentially large number of gateway points for those that represent efficient, but spatially different alternatives. The reason for this is that when a cost surface is non-uniform, the tree has a tendency to form "major branches" along corridors of low cost.

II. Tree Ordering Hierarchies

Strahler stream order (Strahler 1952) is used in hydrology to define stream size using the hierarchy of the tributaries. Based on an earlier stream ordering scheme by Horton (1945), Strahler's ordering modified Horton's to ensure complete objectivity in the structural composition. The ordering method as described by Strahler in his 1952 paper is as follows:

> *The smallest, or "finger-tip", channels constitute the first-order segments. A second-order segment is formed by the junction of any two first-order streams; a third-order segment is formed by the joining of any two second- order streams, etc.*

It is worth noting that other stream order schemes exist. One such scheme is the Shreve stream order (Shreve 1967). This system is simple to understand and has some nice statistical properties in terms of the distribution of order numbers, but seems to have a lesser correlation to the character of actual stream systems. Figure 4 shows an example tree organized by both Strahler and Shreve stream orders.

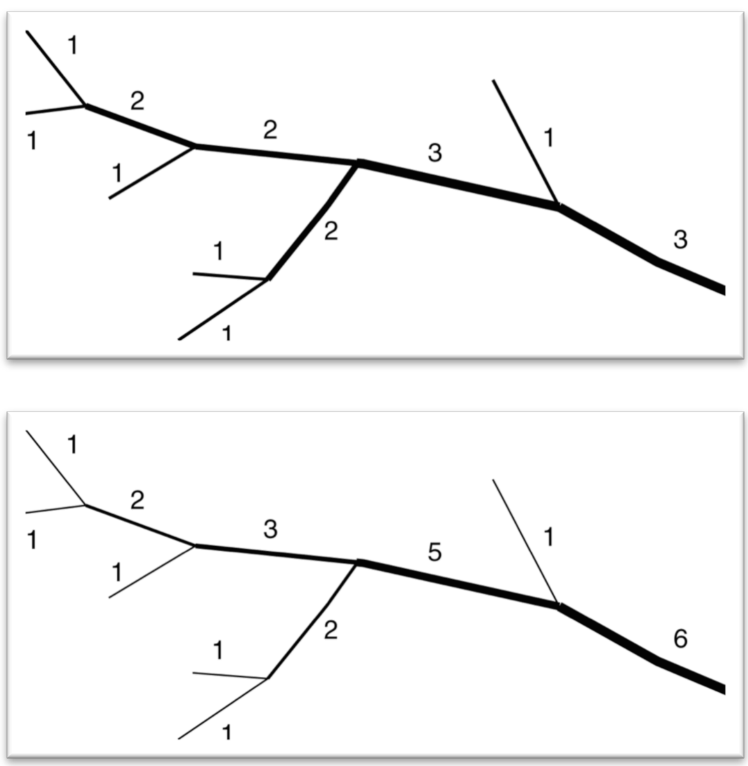


Figure 4. Strahler stream order (top) and Shreve stream order (bottom)
Credit: Wikimedia.org under the GNU Free Documentation License

Fast algorithms for calculating Strahler stream order on a tree have been developed by Lanfear (1990), and Gleyzer *et al*. (2004). Lanfear's approach uses sorting and binary search, and thus runs in approximately O(m_tP*$\log_2(m_t)$) time, where m_t is the number of arcs in the tree and P is the longest path from the "headwaters to mouth". Gleyzer's algorithm uses recursion to dramatically improve the performance of the ordering algorithm, resulting in a method that runs in O(m_t) time. On a graph with n edges and m arcs, the number of edges of a spanning tree $m_t = n–1$, therefore the complexity of the Gleyzer approach is equivalent to O(n). This is much faster than the O((m+n)*log(n)) time required to generate the shortest path trees using our version of Dijkstra's algorithm with a binary heap priority queue. For this reason, we chose to implement Gleyzer's recursive algorithm in order to calculate Strahler order on the shortest path trees. Figure 5 contains an example of the ordering on one such tree. On the left, is a shortest path tree from the lower-left node to all other nodes. The shortest path from the lower-left to the upper-right is highlighted in red. On the right, the tree is re-rendered with the Strahler order denoted both by arc color and arc thickness, where thick/dark arcs are high order and thin/light arcs are low order.

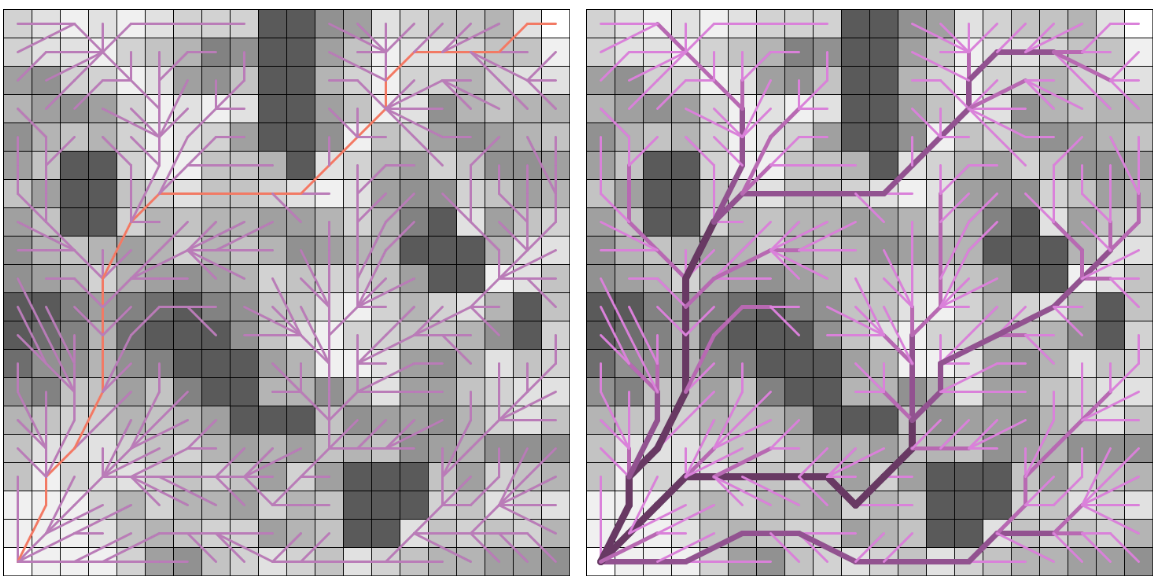


Figure 5. 20x20 Shortest path tree (left), shortest path tree with Strahler order (right)

As discussed in the introduction, a gateway shortest path is generated as the union of the shortest path from an origin to the gateway node, and the shortest path from the destination to the gateway node. All simple gateway shortest paths for all nodes on an undirected graph can be discerned from computing two shortest path trees, one from the origin and one from the destination. Both of these trees may have Strahler ordering applied to them, providing a structural hierarchy to both components of the gateway paths (see Figure 6). Section IV will discuss how Strahler ordering of the shortest path trees can be used for automated selection of "good" gateway points to generate quality shortest path alternatives. First though, the next section will go over evaluation criteria for shortest path alternatives.

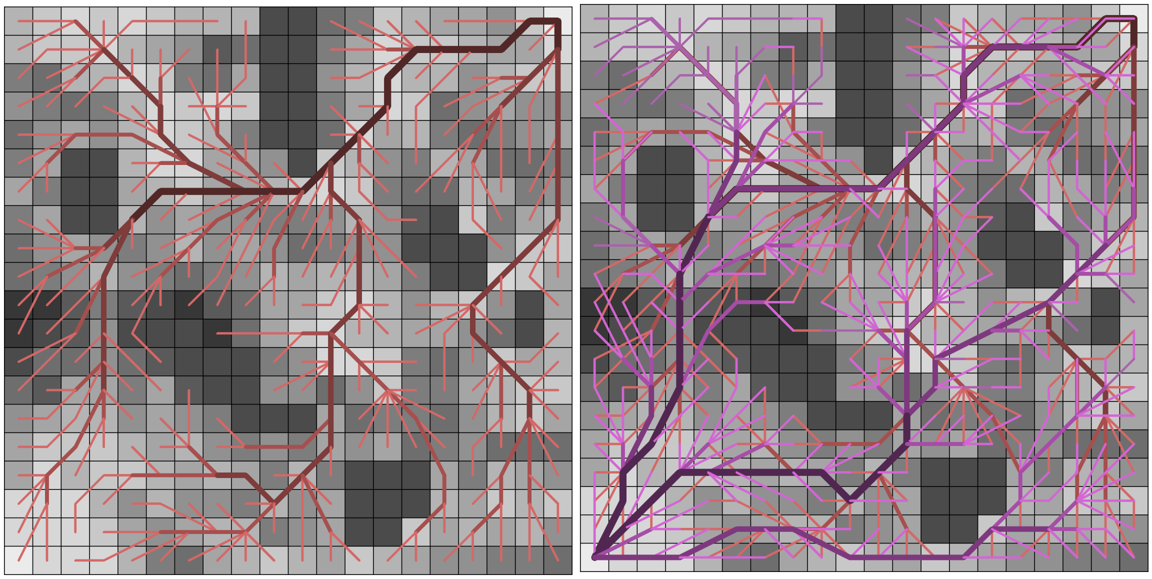


Figure 6. 20x20 reverse tree with Strahler ordering (left), and both trees overlain (right)

III. Evaluating Gateway Paths

Good alternative paths must perform well both in terms of minimizing cost, as well as being spatially different from other paths to which they are being compared. Measuring path cost is simply the sum of the cost of all arcs that compose that path. There are many ways to measure path difference, and in this work we used the area difference metric of comparing the alternate route to the shortest path. This is consistent with the approaches used in Lombard and Church (1993) and Scaparra *et al.* (2014). Figure 7 contains an example of such a path comparison. The image on the left highlights the shortest path by coloring the nodes of that path in orange. The image on the right highlights an alternative path using green arcs, and the area difference is displayed as the red shaded region in between the shortest path and the alternate path.

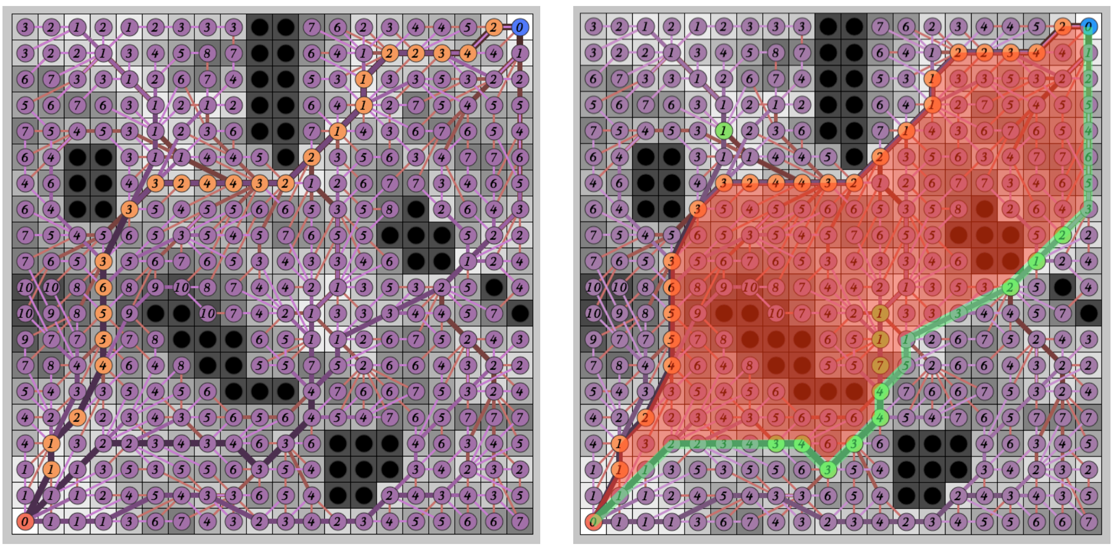


Figure 7. Shortest path (left), alternative path with area difference shaded in red (right)

Lombard and Church (1993) consider the single gateway case, where crossings between the shortest path and the gateway paths are very rare occurrences, and can therefore be disregarded. If no crossings are considered, the area computation reduces to the simple task of using area labels in the shortest path algorithm. Scaparra *et al.* (2014) instead force a gateway path through multiple gateways, in which case crossings are very likely (just consider the simple case of two gateways lying on different sides of the shortest path), and therefore cannot be neglected in the area difference computation. When paths cross, each of the polygons enclosed between the shortest path and the gateway path must be identified and its area calculated. The area computation in this case must be entirely delegated to the gateway path construction phase, since only then can the intersections of the gateway paths with the shortest path be detected. Each newly detected intersection defines a polygon, whose area can be calculated through the formula based on Green's Theorem on the plane. Namely, the area A of a non self-intersecting polygon made up of line segments between M vertices (x_i, y_i), $i = 0$ to M-1, is:

$$A = \frac{1}{2} \left| \sum_{i=0}^{M-1} (x_i y_{i+1} - x_{i+1} y_i) \right| \tag{1}$$

In the area formula, the last vertex (x_M,y_M) is assumed to be the same as the first. Each vertex of the polygon is either a network node or an intersection point between the shortest path and the gateway path. While our approach here uses single gateway shortest paths, we have chosen to use the Green's Theorem approach for area computation, which is more precise and would apply to any future expansion to multi-gateway applications.

Evaluating alternatives is essentially a multiobjective task, as it is desired for alternatives to be both spatially different from the shortest path as well as low in objective cost. One can characterize the performance of gateway paths by plotting a point for each path in objective space, where these two competing objectives are the two axes (path length or cost and spatial difference). Figure 8 is an example of one such plot, depicting the performance of all single gateway paths from a network generated from an 80x80 subset of the Maryland Automated Geographic Information (MAGI) database. The x-axis of the plot measures area difference between the path alternative and the shortest path, and the y-axis plots the objective cost of the path alternative. Notice that the y-axis uses a reverse scale, so that both objectives are improved by moving away from the lower-left corner.

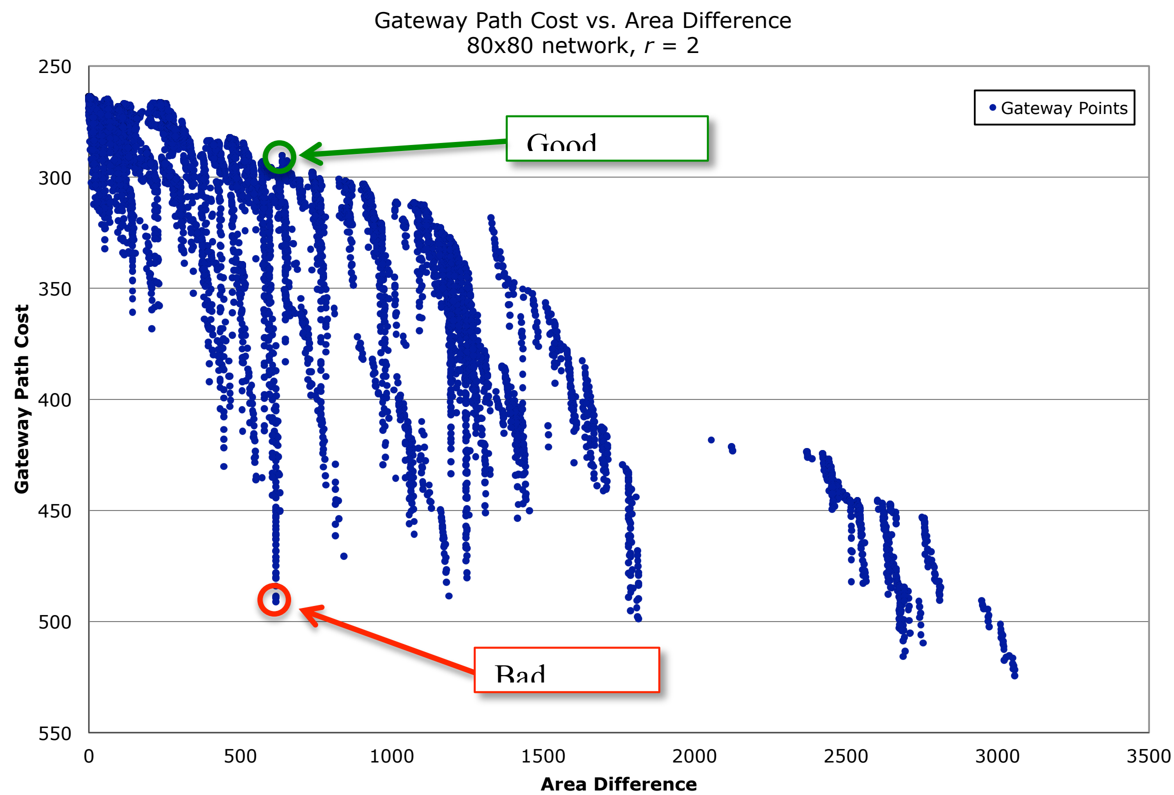


Figure 8. Objective space evaluation of gateway shortest paths

This plot allows one to compare the performance of a selected alternative from all other gateway path alternatives. For example, in Figure 8, the path represented by the point within the red circle would be considered a bad alternative. While it is spatially different from the shortest path, there exist numerous other gateway paths that have the same level of spatial difference but have a better (lower) objective cost. On the other hand, the path represented by the point in the green circle would be a good alternative, as it too is spatially different from the shortest path, but is also among the best in objective cost performance of those paths with similar area difference. In general, the best alternatives will lie on or near the top "ridge" of all the solutions in the objective space plot depicted in Figure 8.

IV. Strahler Threshold Automated Alternative Path Selection

Using the above criteria for evaluating shortest path alternatives, now the question that arises is how do we automate the selection of a set of quality shortest path alternatives? Church *et al.* (1992) discuss using an interactive interface for being able to select a gateway point, view the gateway path, and also view its performance relative to other gateway paths in objective space. While we agree that exploration of path alternatives should certainly include an interactive component, present-day maps are often too large to be able to evaluate all alternatives in an interactive fashion. Even a small map such as the 80x80 example (see Figure 8) generates many hundreds of unique paths that would be cumbersome to evaluate interactively. Instead, it would be useful for the ability to have a model to suggest a small subset of these paths as worth highlighting for closer analysis.

Strahler ordering enables the selection of this subset of quality paths. The idea behind it is that the paths on a shortest path tree will have a tendency to follow low-cost corridors in the data, diverging from those corridors only as necessary to reach all destinations in the network (a complete shortest path tree must cover all nodes in a network with a minimum cost tree, as measured as the sum of the cost over all nodes in the network of the path from the origin to each node). This tendency will result in major branches in the tree that will have high order when analyzed via Strahler ordering. The intersection of high-order branches from both the forward shortest path tree and the reverse shortest path tree would then indicate that a gateway path composed of those high order branches would be a low-cost alternative path. Additionally, given the inherent spacing of these large branches (one cannot have a large branch without many smaller branches feeding into it) there is some assurance that they will be among the set of spatially diverse alternatives.

Strahler stream ordering is an arc attribute, while gateway paths are typically defined by selecting a gateway point. Gateway paths may also be defined by gateway arcs (Katoh *et al.* 1982, Medrano and Church 2014), but shortest path trees do not share all of the same arcs, and thus are not suited for gateway arc path selection. To convert the Strahler arc attribute to a node attribute, we define each node order as the maximum order arc that has an endpoint at that node. Thus each node receives two Strahler order attributes, one for the forward shortest path tree, and one for the reverse.

With these attributes, criteria can be defined to highlight nodes as possible alternatives. The simplest criterion is to highlight all nodes where both forward and reverse labels are greater than or equal to a specified threshold value of t.

V. Computational Experiments

This approach was coded in the Java programming language, using the Processing API (www.processing.org) to help in visualizing the graph and algorithm results. We ran experiments on two networks used in the literature (Lombard and Church 1993, Scaparra *et al.* 2014): a 20x20 manually fabricated raster and an 80x80 subset of the Maryland Automated Geographic Information (MAGI) database. First we will discuss the results on the smaller 20x20 network, followed by a discussion of the larger 80x80 application. In the figures depicting results from the 20x20 network (Figure 9 to Figure 12), the numbers inside nodes (circles) in the decision space network represent the cost or impact of the cell in calculating the traversal impact. The 80x80 network depictions do not display numerical cost values due to display size restrictions.

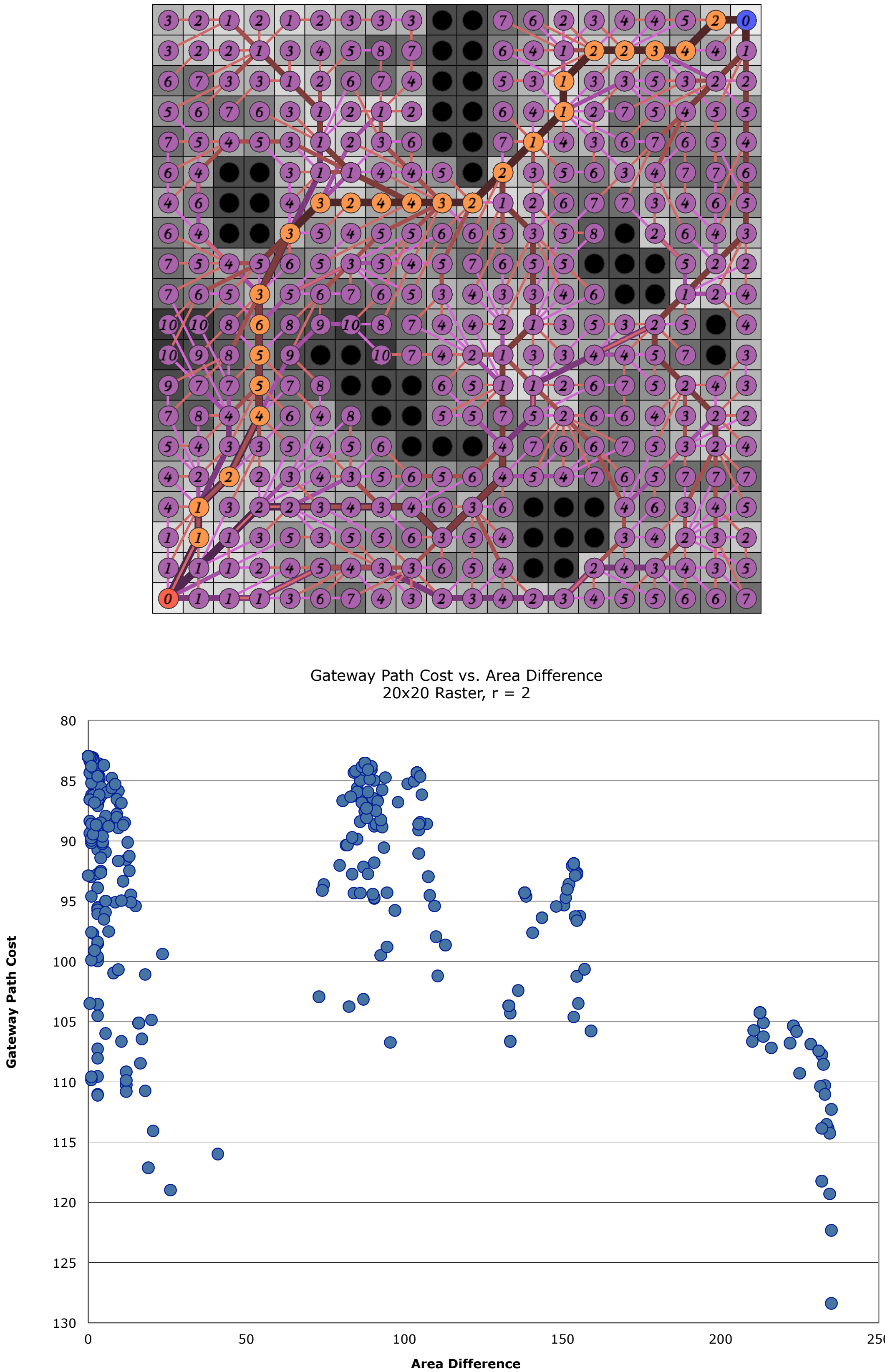


Figure 9. 20x20 r = 2 network all gateway paths: decision space (top) objective space (bottom)

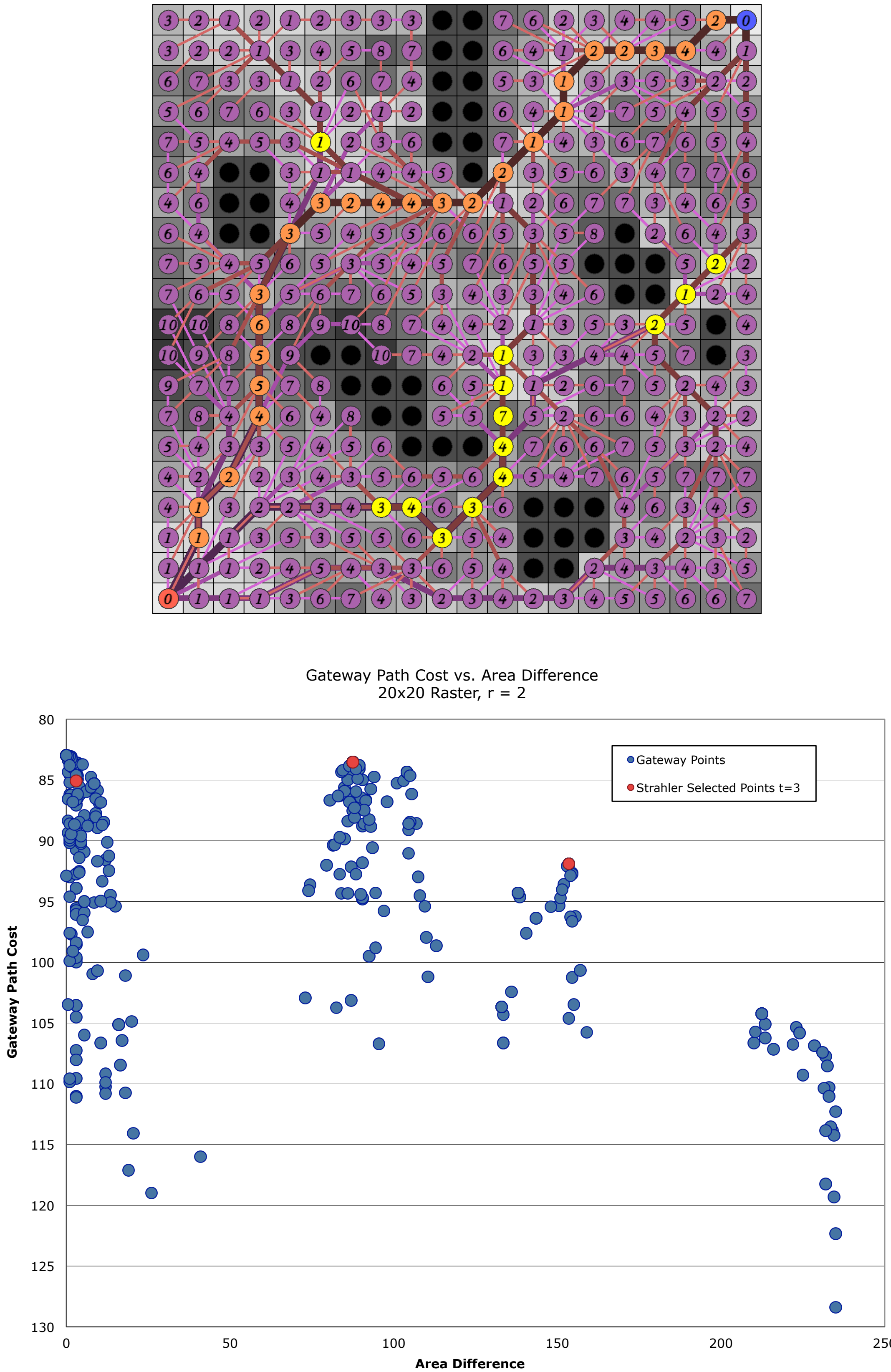


Figure 10. 20x20 network t = 3 gateways: decision space (top) objective space (bottom)

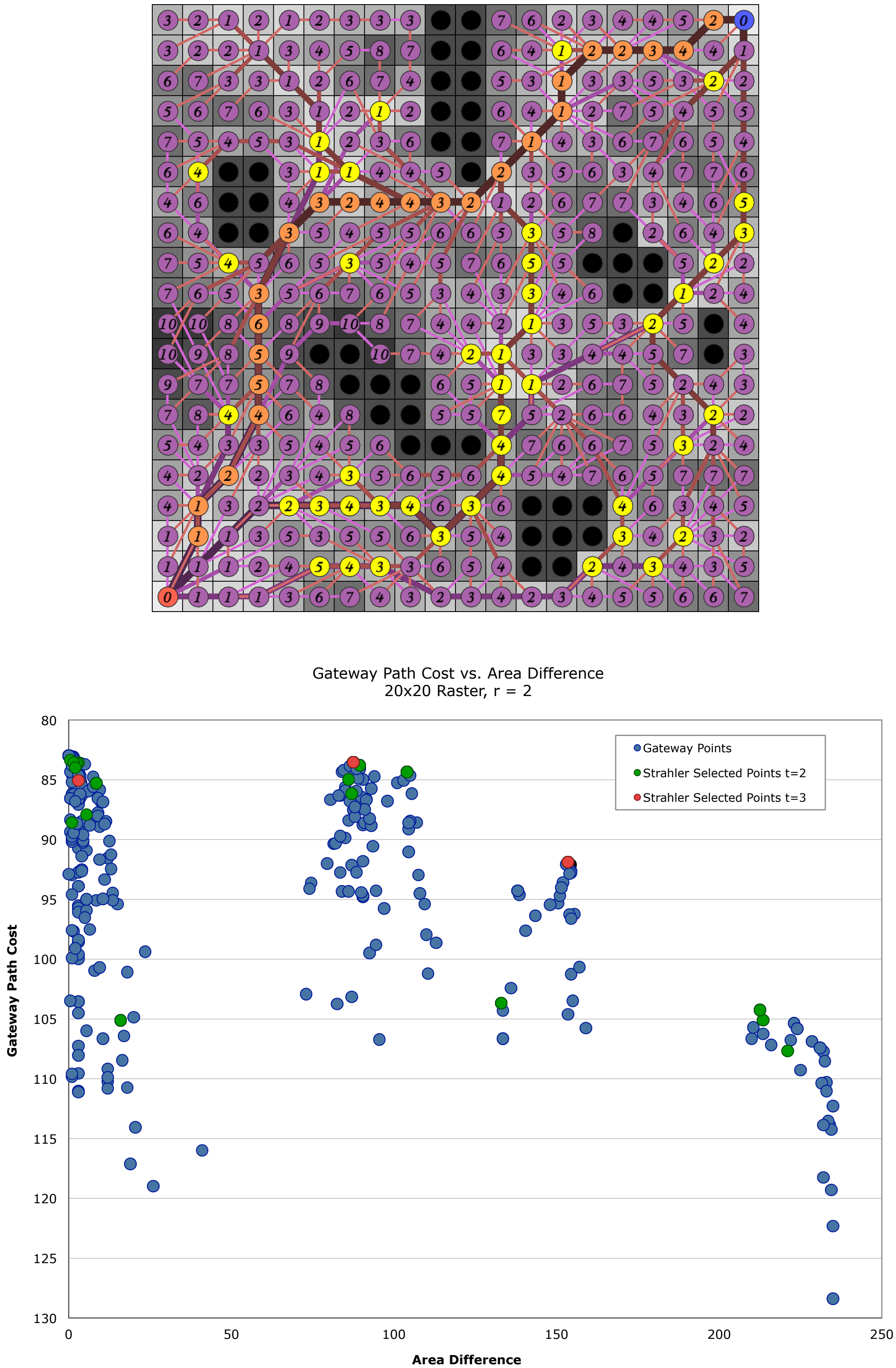
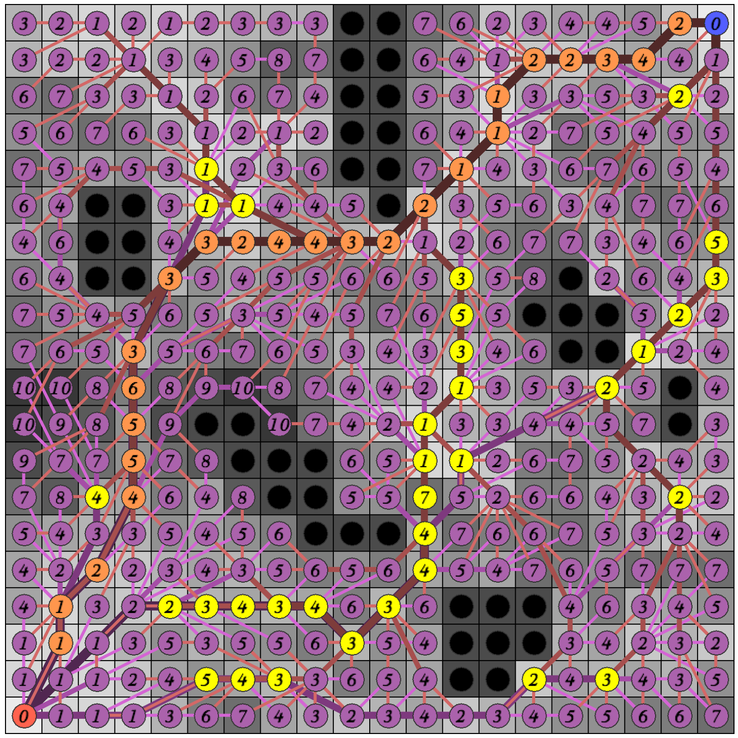


Figure 11. 20x20 network t = 2 gateways: decision space (top) objective space (bottom)



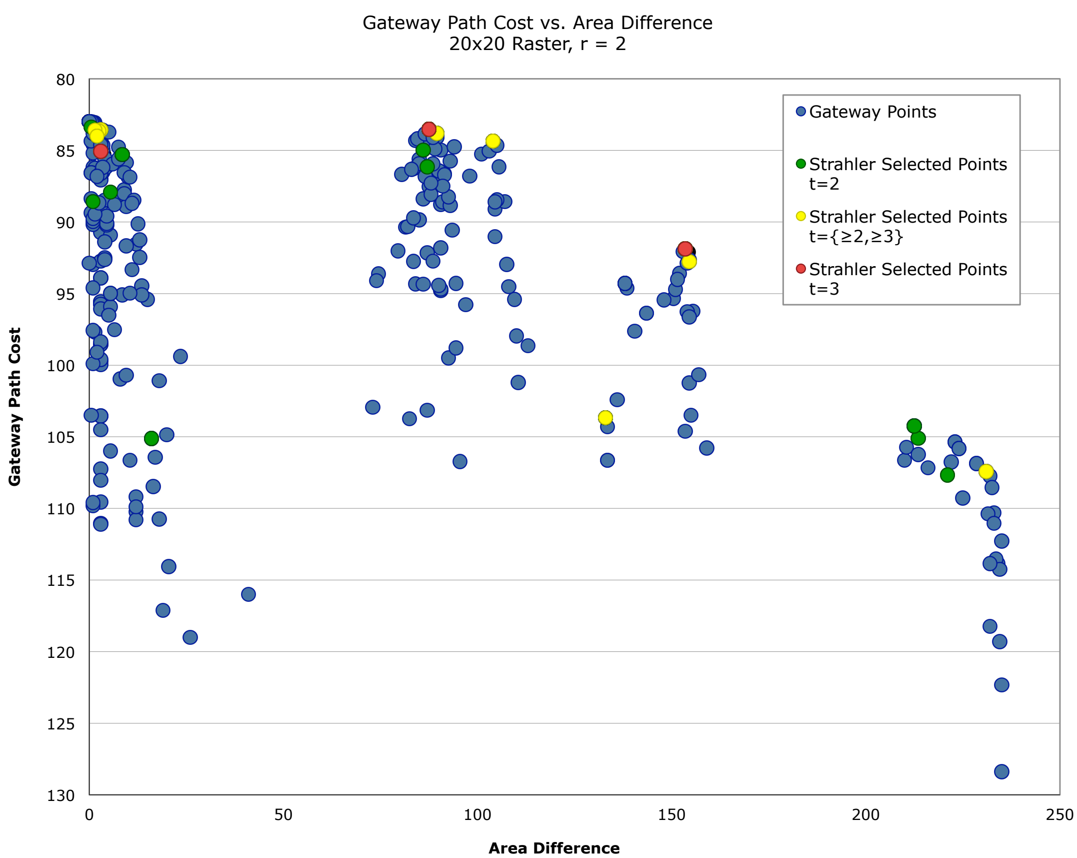


Figure 12. 20x20 network t = {2, 3} gateways: decision space (top) objective space (bottom)

After applying the Strahler ordering on the 20x20, both the forward tree and reverse tree had branches with orders ranging from 1 to 4. Calculating the shortest path tree and applying Strahler ordering to the tree were computed instantaneously on such a small network. Figure 9 first shows the 20x20 network decision space (top) and the corresponding objective space (bottom). The bottom portion of Figure 9 displays the performance of all gateway paths with regards to cost and area difference. Figure 10 highlights the gateway nodes where the Strahler Stream Order (SSO) values of both the forward and reverse shortest path trees at that node are ≥ 3. This results in 13 gateway points being highlighted that represent 3 unique and different paths. These three paths are shown in red in the associated objective space plot. Two of the paths perform extremely well, in that they are both low in cost and also spatially different from the shortest path. The third path (represented by the most upper-left gateway point), is a small deviation from the shortest path. While it too is a low cost path, the deviation from the shortest path is minimal. Lowering the SSO threshold to ≥ 2 for each shortest path tree highlights more alternatives, as displayed in Figure 11. This highlights 44 gateway points, resulting in 20 unique gateway paths (including the three where the SSO in each direction is ≥ 3). These paths are shown in the objective space by green points alongside the $t = 3$ paths in red. More of the Pareto optimal paths have been selected by this lowered SSO threshold value, but more dominated paths have been highlighted as well. Perhaps some other compromise threshold might be able to maintain most of the non-dominated solutions while eliminating most of the dominated solutions. Figure 12 displays the results when trying the criteria of selecting nodes with one Strahler label ≥ 2, while the other must be ≥ 3. This combination of values falls between the stringency of each SS0 ≥ 3 and each SSO ≥ 2, and results in 33 nodes highlighted that define 12 unique paths. In the objective space chart, these paths are colored yellow. In this case, all but one of the paths are on or near the Pareto frontier of solutions, while missing only 1 or 2 Pareto paths on the far right of the objective space chart that were caught by the $t \geq 2$ threshold (green).

Next we tested this method on the 80x80 $r = 2$ MAGI network. After applying the Strahler stream ordering on this network, the forward tree had branches with orders ranging from 1 to 7, while the reverse tree had branches with ranges from 1 to 6. Calculating a shortest path tree on this network took about 44 milliseconds on a Macbook Pro laptop, while the Strahler stream ordering was accomplished in 7 milliseconds. Figure 13 displays images of the forward and reverse Strahler ordered trees for the 80x80 MAGI network.

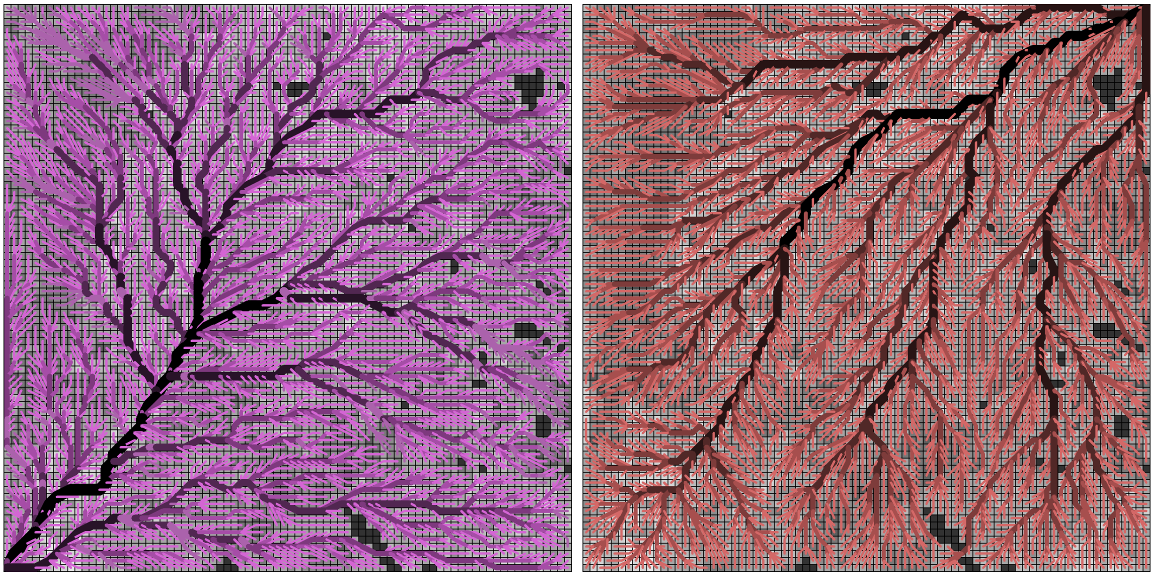


Figure 13. 80x80 shortest path trees with Strahler ordering: forward tree (left) and reverse tree (right)

For the automated path selection, there is only one path for a threshold of t = 5 in both directions. This gateway point is highlighted in green in the decision space and the objective space in Figure 14. This gateway path is near-Pareto optimal from among the gateway possibilities, and represents a reasonably good path alternative.

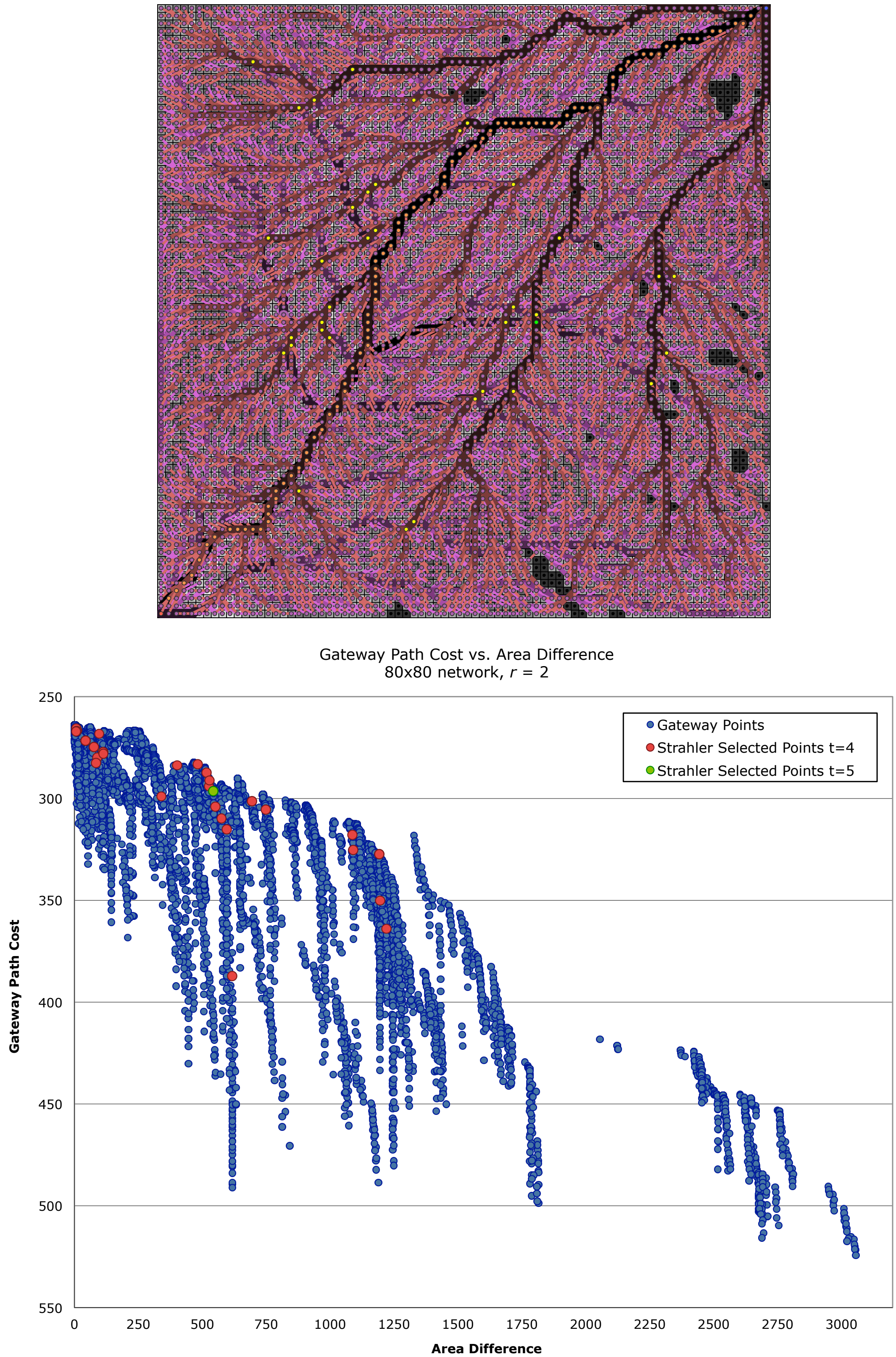
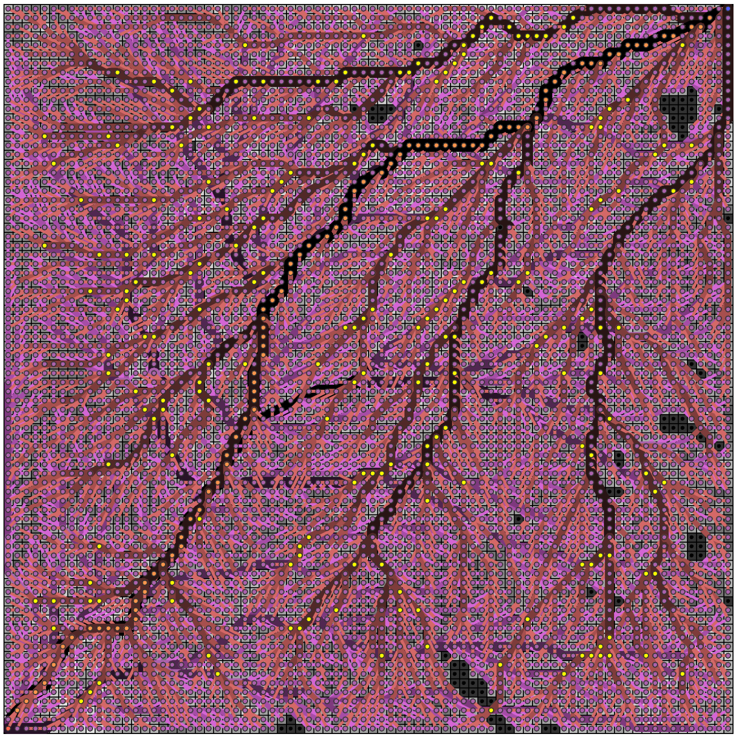


Figure 14. 80x80 network t = 5 & t = 4 gateways: decision space (top) objective space (bottom). The t = 5 gateway point in decision space is highlighted in green.



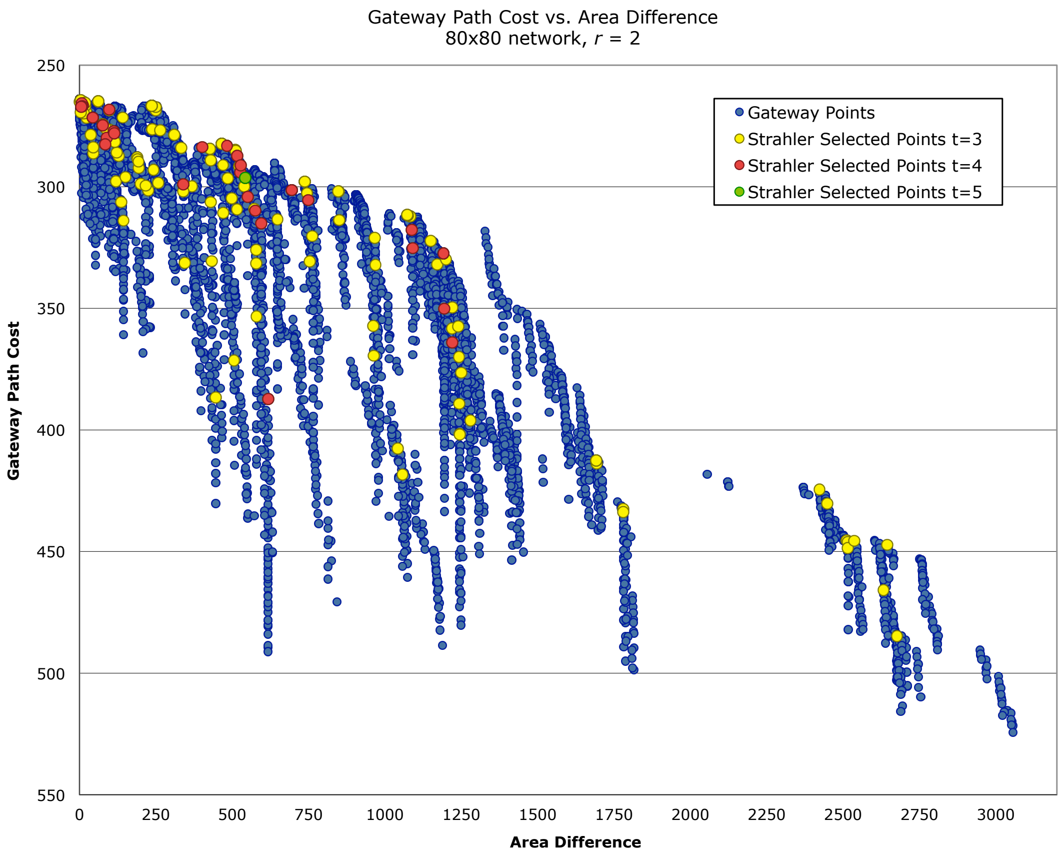


Figure 15. 80x80 network t = 3 gateways: decision space (top) objective space (bottom)

Figure 14 also highlights the points and associated paths identified by using a SSO threshold standard of $t \geq 4$. These consist of 37 gateway points that represent 27 unique gateway paths. About half of these paths are on or near the Pareto frontier, while the other half would be considered inferior. They also do not reach some of the larger area difference regions of the objective space, indicating that a less stringent threshold would be required in order to select paths of higher spatial diversity from the shortest path.

When using the less stringent SSO threshold of $t \geq 3$ for each tree as displayed in Figure 15, 203 graph nodes are highlighted. This resulted in many more unique gateway path options being identified. These gateway paths cover more of the Pareto optimal frontier in objective space, filling in many of the voids that were missed by the SSO $t \geq 4$ threshold. Additionally, much higher spatial diversity is achieved, with paths now being selected in the second tier of area difference on the far right of the objective space chart. Almost all of the higher area difference paths are on the Pareto frontier, and thus these paths would be considered excellent alternatives. On the other hand, many dominated paths were selected within the region of smaller area difference solutions.

VI. Concluding Remarks

We used Strahler stream ordering to define a hierarchy for the shortest path trees in order to aid in selecting good gateway points from among all the gateway point possibilities. Each node in a network was given two Strahler stream order values based on the order hierarchy of the forward and reverse shortest path trees. Sets of path alternatives were selected by choosing gateway points with Strahler stream orders that exceeded a given threshold. These path alternatives were evaluated based upon two criteria: minimizing the path cost, as well as maximizing the path area difference relative to the shortest path.

Computation of these alternatives was extremely efficient, where the most time consuming portion of the calculation was in the generation of the shortest path tree, which has been shown to be a polynomial computation, and has an extensive literature on efficient algorithms. With Strahler stream ordering and path filtering being linear operations, this method is very fast at heuristically selecting a set of path alternatives.

On the 20x20 network, the Strahler stream order based selected alternative path sets showed excellent performance in as measured in path length vs. shortest path area difference. In the 80x80 data set, the selected alternatives also performed well for the stringent thresholds, but suffered when the threshold became too unrestrictive. The low SSO $t \geq 3$ threshold also selected far too many paths to be useful in a realtime analysis, thus indicating that it is important to select a threshold criteria suitable to the data being analyzed. It is important that this method be used interactively:

1) To ensure proper threshold values are being applied for the given data. Using this approach on various data sets will result in different magnitudes of Strahler order numbers as the size and character of the data change
2) To verify the quality of the solutions. The area difference metric is used to only compare a path alternative to the shortest path, and not to other path alternatives. Two paths with different routings in decision space may have associated points that lie very near each other in objective space.

This research was explored only within the context of selecting corridors over terrain networks. Future work should extend to other network types, using this for alternative route applications where speed of acquiring results is paramount. For example, road networks have an intrinsic hierarchy consisting of residential streets to main roads to interstate highways, and could be particularly suitable to the selection of alternatives using a hierarchical threshold.

References

Church, R.L., S.R. Loban & K. Lombard, (1992). An interface for exploring spatial alternatives for a corridor location problem. *Computers & Geosciences,* 18, 1095-1105.

Gleyzer, A., M. Denisyuk, A. Rimmer & Y. Salingar, (2004). A fast recursive gis algorithm for computing strahler stream order in braided and nonbraided networks. *Journal of the American Water Resources Association,* 40, 937-946.

Horton, R.E., (1945). Erosional development of streams and their drainage basins; hydrophysical approach to quantitative morphology. *Geological Society of America Bulletin,* 56, 275.

Katoh, N., T. Ibaraki & H. Mine, (1982). An efficient algorithm for k shortest simple paths. *Networks,* 12, 411-427.

Lanfear, K.J., (1990). A fast algorithm for automatically computing strahler stream order. *Water Resources Bulletin,* 26, 977-981.

Lombard, K. & R. Church, (1993). The gateway shortest path problem: Generating alternative routes for a corridor location problem. *Geographical Systems,* 1, 25-45.

Medrano, F.A. & R.L. Church, (2014). Corridor location for infrastructure development: A fast bi-objective shortest path method for approximating the pareto frontier. *International Regional Science Review,* 37, 129-148.

Scaparra, M.P., R.L. Church & F.A. Medrano, (2014). Corridor location: The multi-gateway shortest path model. *Journal of Geographical Systems* 16, 287-309.

Shreve, R.L., (1967). Infinite topologically random channel networks. *The Journal of Geology,* 178-186.

Strahler, A.N., (1952). Hypsometric (area-altitude) analysis of erosional topography. *Geological Society of America Bulletin,* 63, 1117.