The Philadelphia Districting Contest: Designing Territories for City Council Based Upon the 2010 Census

Ram Gopalan
School of Business, Rutgers University, Camden, New Jersey 08102, ram.gopalan@rutgers.edu

Steven O. Kimbrough
Operations and Information Management, The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania 19104,
kimbrough@wharton.upenn.edu

Frederic H. Murphy
Marketing and Supply Chain Management, Fox School of Business, Temple University, Philadelphia, Pennsylvania 19122,
fmurphy@temple.edu

Nicholas Quintus
Office of Watersheds, Philadelphia Water Department, Philadelphia, Pennsylvania 19107, nquintus@temple.edu

The City of Philadelphia recently redesigned districts for its council members based upon the 2010 census. The districting process evinced considerable public interest and engagement because council districts from the prior census had significant shortcomings. During the 2010 redistricting process, several public interest groups came together to organize a districting contest. The organizers hoped to increase public engagement in the districting process and to proactively offer several well-constructed examples of city council districts that minimize gerrymandering. We were active participants in the contest, developing methodologies for finding good solutions to large integer programs that enabled us to win in one contest category and make presentations before the city council in this and another category. This article describes the unfolding of various events surrounding the districting process, the methodologies we developed, and the influence that the contest ultimately had on the design of city council districts.

Key words: political districting; optimization; genetic algorithms; games.

Once every 10 years, new U.S. census numbers trigger legislative redistricting at the national, state, and local levels. Because this cycle occurs regularly, we decided to develop in advance an optimization model using data from the 2000 census; this model would let us produce districts as soon as the new census numbers became available. It would also allow us to demonstrate to Philadelphia's city council what good districts *should* look like.

Our goal was to improve on the existing, poorly-designed districts through public engagement. According to a study comparing districts in cities throughout the country from 2002 to 2006 (*Azavea* 2006), although Philadelphia had only 10 districts in the city, it had two of the most gerrymandered city council districts in the United States. Districts 5

and 7 were no wider than a roadway in several places (see Figure 1).

District 5 was designed to be predominantly poor and minority, but included a dangling portion at the bottom that included a piece of one of the wealthiest neighborhoods in the city. The addition of this piece gave the incumbent both a population that would vote for him and constituents from whom he could raise a substantial amount of money. District 7 was held by the lone Hispanic council member; she wanted a more compact district with a larger proportion of Hispanics than the one she had inherited.

Our intention in developing our model was not to consult as insiders, but to take advantage of the increase in PC computing power and inexpensive optimization software to develop a professional-grade
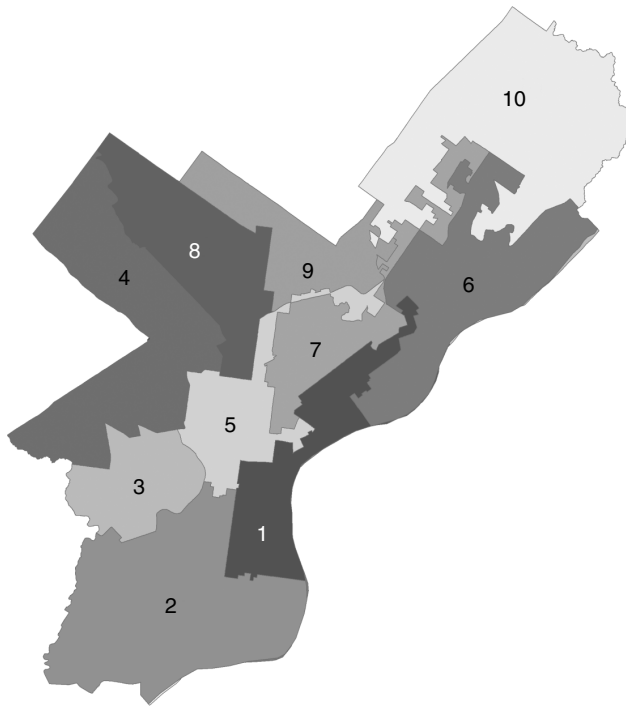
Figure 1: This map of Philadelphia's 10 city council districts prior to the 2011 redistricting shows the extensive gerrymandering in Districts 5 and 7.

model that would demonstrate to a broad audience what nongerrymandered districts would look like. Optimizing voter districts involves building large integer programs with publicly available data that would allow us to experiment with and use research ideas for sensitivity analysis of integer programming (IP) solutions on a real problem.

Any good solution to the redistricting problem must satisfy three conditions: contiguity, compactness, and equal population. In a contiguous district, a path connects every political subunit in the district to every other subunit, staying within the district. Compactness has several definitions based on the underlying notion that the political subunits do not sprawl over a map, such as the salamander-shaped district that led to the coining of the word gerrymander. Equal population is usually taken to mean that all district populations must lie within a given percentage of a target value (with the target value equal to the total population of a political unit, such as a state or a city, divided by the number of required districts).

Redistricting models have taken three forms: extensions of the facility location model (p-median model); a set partitioning formulation, where columns correspond to combinations of basic population units that represent feasible districts, analogous to the crew-scheduling models of airlines; and heuristic approaches, such as tabu search and genetic algorithms. Murphy et al. (2013) include a survey. Hess et al. (1965) were the first to use operations research (OR) techniques in this context. They combined optimization with a heuristic to redistrict the state of Delaware. Building on the facility-location literature, Garfinkel and Nemhauser (1970) developed a model where they enumerated all possible assignments of political units to districts that meet constraints on population and that are also contiguous. They specified a compactness measure that takes the ratio of the maximum distance between political units in the district and the area of the district. They then solved a set-partitioning problem to cover all population units with the requisite number of districts and minimized the maximum value of their compactness measure. However, they had difficulty solving problems with 50 or more population units, given the computers available then. Mehrotra et al. (1998) developed a branch-and-price extension of the original Garfinkel and Nemhauser model.

The major impediment to building a model to solve the complete districting problem has been to find a way to implement the contiguity condition. Hess et al. (1965) rejected noncontiguous solutions generated by their heuristic. Mehrotra et al. (1998) stated that imposing contiguity directly in the model would entail an exponential number of constraints, which is why they first generated the set of all possible contiguous districts. Contiguity was also an important issue considered by Bozkaya et al. (2003), who developed a tabu search approach in which all solution changes in the search had to maintain contiguity. More recently, Bozkaya et al. (2011) designed electoral districts for the city of Edmonton in Canada using their tabu-search approach, a remarkable statement about the quality of the political leadership in that city.

Two papers have used graphs to solve the problem of combinatorial explosion when imposing contiguity constraints. Williams (2002) applied the contiguity

constraints to trees defined over primal-dual planar graphs. Shirabe (2009) built on work by Zoltners and Sinha (1983) to develop a simpler method that measures flows in a graph defined using the population units as nodes. He required that two units with positive flow between them be in the same district and that each district have a designated root node that serves as an anchor node for the district.

Developing a Districting Model

We chose to base the first version of our model on the facility location model. We formulated a model in which each of the 155 neighborhoods in Philadelphia could potentially be the center of one of the ten political districts. The objective function was the sum of the squared distances from each population unit (neighborhood) in a district to the center of the district. At our request, the geospatial software developer Azavea (http://www.azavea.com) constructed a data set containing the populations and centroids of the 155 Philadelphia neighborhoods. We were able to establish contact with Azavea through a local political activist, and used these data to begin our IP modeling. As in Hess et al. (1965), the neighborhoods chosen as the centers are the equivalent of warehouses, and the neighborhoods assigned to a district are the service territory. The ten districts are constrained to be within a given percentage of 1/10th of the city's total population. The model details are in Appendices A and B.

We showed a map of our optimal districts to the activist who introduced us to Azavea and who knew the Philadelphia political and demographic landscape. He pointed out the obvious flaws in our solution. Philadelphia is a city of neighborhoods and its residents have a strong neighborhood identity. Consequently, a districting plan should consider neighborhood sensitivities. He pointed out that districts should not cross the Schuylkill River, which divides most of the city from West Philadelphia, or certain sections of Roosevelt Boulevard, a major road that goes from the river through the northeast portion of the city (or the crossings should be minimized to construct districts with coherent identities). As is always the case, an optimal solution is optimal in terms of the specified objective, not necessarily in terms of the real-world problem to be solved.

Our first response was to impose additional constraints that introduced these neighborhood barriers into the model and to generate new solutions. We were not able to impose Roosevelt Boulevard as a barrier for its whole length and meet the population-target constraints. We then retained Roosevelt Boulevard as a barrier in the older, walking neighborhoods and did not make it a barrier in the newer car-oriented neighborhoods in the far northeast. To improve our solutions, we obtained data on the city's roughly 1,800 census block groups, because this population unit has finer granularity than the neighborhoods, allowing us to produce more compact solutions. However, using census block groups greatly increased the size of the IP problem. To keep the size of the model under control, we continued to use neighborhoods for (potential) district centers.

The Contest

With our model working, the next step was to determine how to get publicity for our results and maybe have a little influence on the districting decision. We began to use our personal networks to connect to local reporters and foundations. In early July 2011, several organizations, including Azavea, local public radio and television stations, the University of Pennsylvania Center for Civic Engagement, and a local newspaper, announced a contest with prizes for constructing the best districts according to several quantitative criteria. They held an organizational meeting in late July and about 150 people showed up. We realized that winning the contest could be our vehicle for publicizing our results and advancing our goal of challenging the city council to do a better job than it had previously.

The good news was that we already had a working model for demonstrating our capabilities and providing solutions, given that the contest deadline was little more than a month away. The bad news was that none of the contest's quantitative criteria for judging solutions (i.e., the contest organizers' objective functions) matched the objective function that we had been using. Their criteria consisted of (1) the minimum population spread for the solution, defined as the maximum of the ten district populations minus the minimum of the ten district populations; (2) the

fewest splits between council districts of the 66 wards, their proxy for neighborhoods; and (3) the most compact districts as defined by the Schwartzberg measure, which is the sum over the districts of the boundary lengths of each district divided by the circumference of the circle with the same area as the district.

Wards are political units for vote tallying. Each political party elects one or more captains for each ward. Because the wards were last changed in the mid-1990s, they do not really match the neighborhood definitions. However, they match the political geography of the city and are a useful proxy for showing the consequences of keeping neighborhoods together, which is important in older cities in which neighborhood identity is deeply ingrained in the residents. The most fine-grained geographic unit used in the contest was the ward division, a subunit of wards. Each ward has between 10 and 50 divisions per ward, leading to 1,687 geographic units that needed to be assigned. For our optimization model, we retained the original neighborhood centroids as potential district centers and used wards and ward divisions as appropriate for the population units to be assigned to districts.

Our problem was the multiplicity of definitions of compactness and a contest definition that did not match ours. We had been using the sum of squared distances from district centroids; their criterion was to minimize the Schwartzberg measure. Minimizing squared distance tends to minimize the maximum distance across districts, the district diameter. Focusing on boundary length tends to value smooth boundaries at the expense of the maximum distance across districts. However, no best definition of compactness exists. Young (1988) points out the flaws of the various criteria, showing that no easy answer exists on what constitutes the best criterion for compactness. The contest organizers chose a criterion that geographers like, whereas we started with one that operations researchers like, and we were forced to adapt to their criterion with limited time.

A contest open to the general public requires geographic information system tools that a nonexpert can use. Azavea had received a grant from the Alfred P. Sloan Foundation to develop a Web-based tool to allow contestants to form districts, using mouse clicks, from geographic subunits (in this case ward divisions) that define the territories for voting locations. The web access to its software for the contest was through a website (http://fixphillydistricts.com) that was still operational as we were writing this article. Azavea also provided a spreadsheet with the populations and geographical coordinates of the ward divisions for those contestants, like us, who wanted to improvise using their own methodologies.

One feature of Azavea's Web software is that it has a leaderboard where contestants can post their trial solutions. The leaderboard provides the quantitative metrics for posted solutions without actually revealing what the solutions are. Given the interest in local redistricting and the large talent pool in the region, contestants were soon posting their solutions. Several solutions that did not split any wards and met the district-population and contiguity constraints were posted very quickly. Well before the close of the contest, one team (person) posted a solution in which the deviation from the population target was zero, a remarkable achievement given the contiguity constraints and the discrete size of the ward divisions.

The criterion of not splitting wards fit naturally into our modeling approach. All we had to do was solve our original model with the ward-level data, instead of our trial data on neighborhoods, and enforce the integrality of the assignments of wards to districts. Moreover, Philadelphia has only 66 wards, simplifying the computation of an optimal solution. Using neighborhoods as centers and wards as the population units, the model has 4,422 integer variables. Although the Shirabe (2009) formulation of the contiguity constraints that we implemented does not exponentially explode the model size, letting each neighborhood be a centroid and working with ward divisions, the model would have 261,640 integer variables, which was beyond the capabilities of our computers. Therefore, in all cases, we first solved the model without the contiguity constraints. We then took the centroids from that solution and resolved the problem with the centroids fixed and the contiguity constraints imposed, reducing the number of contiguity constraints by a factor of 66, the number of wards. This gave us our first contest entry for the criterion of zero ward splits.

One problem with solutions to optimization models is that they have a take-it-or-leave-it quality.

The black-box optimizer gives an optimal solution to the model and the user is supposed to take this to be the optimal solution to the problem. In linear programming, this problem can be alleviated by exploring around the neighborhood of model parameters using sensitivity analysis. However, solutions resulting from marginal changes in coefficients do not have the same meaning in integer programs, because dramatic shifts in the optimal solution can result from minor changes in parameters.

Given the multiple definitions of compactness, the multiobjective nature of the real problem, and neighborhood interests in being grouped or not grouped with other neighborhoods, humans need some freedom to choose the final solution to implement. Furthermore, if some choices that allowed consideration of the legitimate self-interests of the politicians who are redistricted were available, then an analytic approach would be more palatable to politicians. The degree to which politicians exercise self-interest (and whether they exercise it at the expense of the polity) is the important variable, not whether they are self-interested per se.

We decided to implement a simple approach to explore the solution space around the solutions initially generated using the integer program. We generated alternatives to the optimal solution with metaheuristics by implementing an evolutionary algorithm. Using this algorithm and multiple starting points (four in all) created by varying the objective function in the integer program, we generated 116 legally valid solutions, none of which split the wards. This meant that not only were we in a position to offer solutions that met one of the contest objectives (no ward splitting), we were also in a position to present solutions that were reasonably good at meeting compactness and population-size criteria or, alternatively, were more coherent in their demographics.

In parallel with breeding multiple solutions having zero ward splits, we worked on developing a compact solution using the Schwartzberg measure. We had two choices: we could formulate a new optimization model with this specific objective, or we could take a solution that minimized our original objective, which is the weighted squared distance (WSD) of population units from district centroids, and use heuristics to improve it relative to the Schwartzberg measure.

An optimization model for the Schwartzberg measure would involve an indicator variable for each edge between two districts. This would entail a very large model because it requires a decision variable for each pair of adjacent ward divisions in the set of 1,687 ward divisions. Furthermore, the objective is nonlinear because the measure is a ratio that involves calculating the perimeter of a circle having the district's area. Although we wanted to try to formulate the optimization model, Azavea could not provide a file with the lengths of the boundaries between each pair of ward divisions. Therefore, we resorted to using heuristics, starting from one of our solutions.

We first solved the integer program using ward divisions for the building blocks of the districts and the neighborhoods as the centers. Again, this was a two-step procedure of solving for the centers without the contiguity constraints, and then solving for the districts after fixing the centers. Then we manually applied some simple heuristics, using the contest-provided graphics tools to move from the center-of-gravity solution to our best solution using the Schwartzberg measure.

The rules of thumb that we used are as follows: Having straight boundaries is more important than having the ward divisions as close as possible to the district centers. Diagonals tend to generate shorter total boundary lengths. When a boundary between two districts touches the boundary of a third district, generating a solution such that the boundary between the first two is perpendicular to the boundary of the third shortens the boundary length.

We needed several revisits to the trial solution to come up with the plan that we finally submitted. We kept observing more opportunities to apply these rules and improve the solution by letting the solution sit overnight and then looking for more improvements. As we were working, we watched the leaderboard to see how other contestants were doing with the compactness metric. With the early passes using the heuristics, we came up with a solution that was better than all others that had been posted. We chose a strategy of posting on the leaderboard our second-best solution to set up that solution as the target to beat but not fully reveal our hand. Our second-best solution put us sufficiently far ahead

that other contestants stopped posting improved solutions. We thought that unless some lurkers were not posting their solutions, we had the winning entry in that category.

After we generated our solutions, we had to put together our results for submission. As in all projects with deadlines, we had plenty to do at the last minute. We were generating more solutions for the no-ward-splits category until the very end (the evolutionary algorithm was slow in finding new solutions). We also continued to test for improvements in our compactness solution. In the final hours, we were writing commentaries on our solutions and were in constant communication with each other while working from our homes. We could not get together face-to-face because we were in the middle of Hurricane Irene, hoping we would not lose power. One of our motivations for constantly staying in touch was to ensure that everyone on the team had power and could continue on the project.

Our Results

The contest rules allowed only one submission per team or individual. We chose to be recognized as a team because we had worked so closely together and decided on our compactness solution because it was unique in that category. Submitting only one of our solutions without ward splits did not carry the full import of what we had accomplished in that part of the contest. However, we did write up and make public all of our results. Our results and testimony before the city council on both the compactness and solutions without ward splits are available at http://opim.wharton.upenn.edu/~sok/phillydistricts.

Because the city council did not initially schedule any hearings on redistricting other than one hearing very early in the process, the local civic groups and media started demanding that it hold more hearings. Eventually, it scheduled two additional hearings. At the first hearing, we presented our testimony on the plans that did not split wards. We showed two solutions, one that minimized population deviation and one that came closest to creating neighborhood coherence.
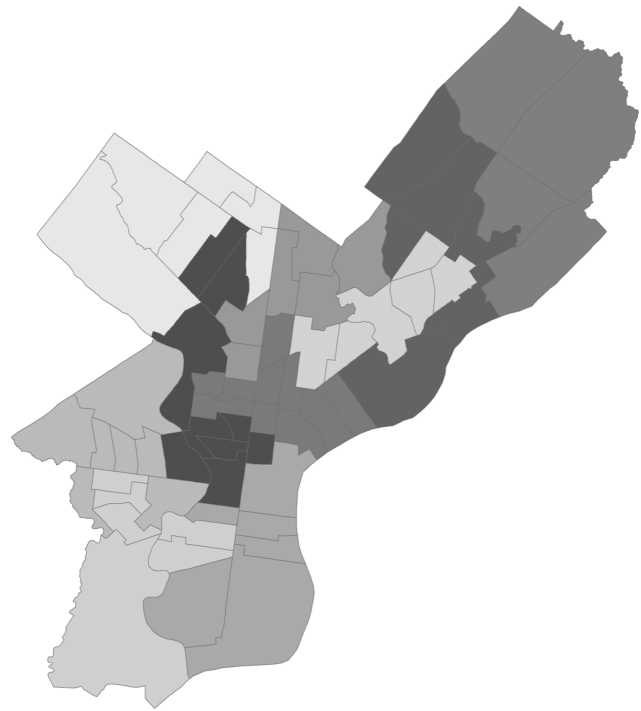

Figure 2: This plan minimizes population deviation (= 2,335) **with zero ward splits.**

The plan that minimizes population deviation (see Figure 2) is clearly ungainly because the resulting districts are not compact and look as if they are gerrymandered, although the solution is purely analytic. The plan we highlighted in the testimony (see Figure 3) is the most neighborhood-friendly one we found, based on judgments from colleagues with deep knowledge of Philadelphia.

Using the Schwartzberg measure as implemented in the contest (the contest software used the inverse of the original measure with boundary length as the denominator), this plan has a value of 0.6263 versus the 0.7513 for our submitted plan in the compactness category. In addition to having a good Schwartzberg measure for compactness, the plan also succeeds by having low distances from one end of a district to another. The shorter distances allow city council members to deal with a smaller geographic spread than the current city council map does. This plan provides the most appropriately shaped districts possible without splitting wards. The equipopulation score is 5,804, roughly double the equipopulation score for
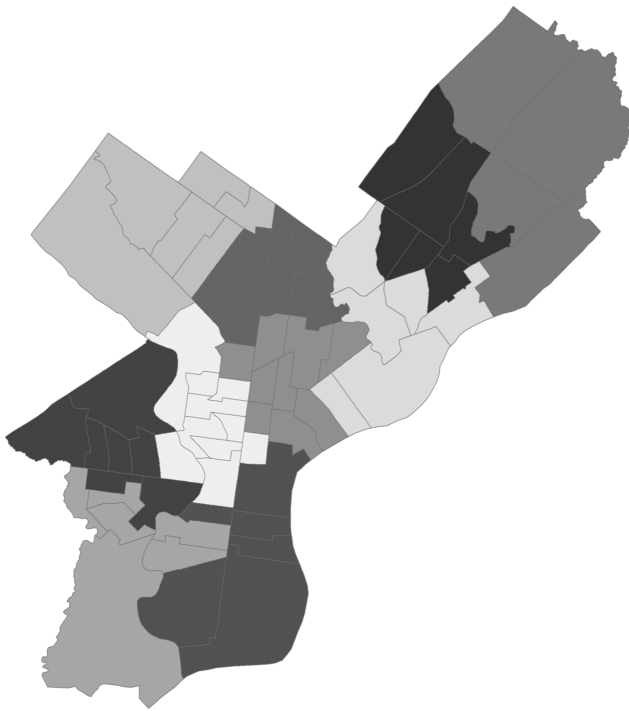
Figure 3: This plan maintains neighborhood coherence and keeps the wards intact.
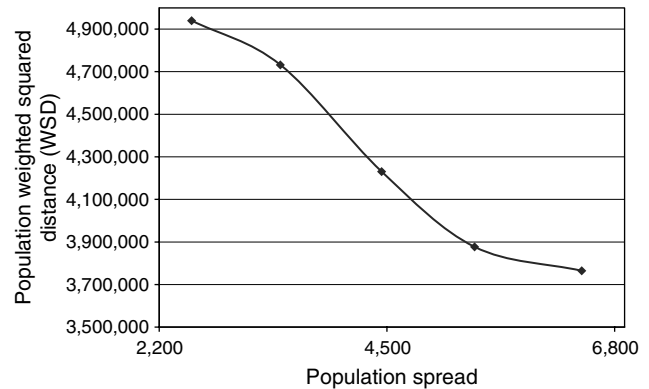

Figure 4: This trade-off curve between population WSD and population spread shows the trade-off between the tolerance for equal population and compactness.

our solution that minimizes population deviation, but falls within the permitted five percent deviation from the target population for each district, as stipulated by the contest rules. (The Philadelphia courts have accepted a ten percent limit, but the conventional wisdom is that anything worse than five percent is legally at risk.) Nevertheless, the boundaries do not look very good: natural barriers, such as rivers and highways, are crossed and some districts have wide disparities in income and other demographic characteristics.

We decided to look at the trade-off between compactness using our original definition and the maximum population deviation (see Figure 4).

We include this chart to illustrate that defining good districts has meaningful trade-offs. The metric population WSD is the sum of the WSDs between the wards and their district centers. Population spread is the population difference between the districts with the largest and smallest populations. A clear trade-off exists between these two measures. The actual shape of the curve in Figure 4 is not really a logistic curve. The smallest population spread that was feasible

without splitting the wards was roughly 2,300. In the testimony, we offered to generate, at no cost, large numbers of plans for any definitions of neighborhoods that the city council chose. Needless to say, our offer was not accepted.

Figure 5 depicts our winning plan for compactness. Our compactness score was several percentage points above the score of the next-best solution in the leaderboard, which shows the power of a good starting solution, reasonable heuristics, and the willingness to revisit trial solutions to garner further improvement. Although our plan is extremely compact, it is not a good redistricting plan as it currently stands, because it shows the weakness of relying solely on quantitative measures to rate plans. It splits too many neighborhoods, showing the weakness of the Schwartzberg measure of compactness when confronted by Philadelphia's oddly-shaped ward divisions.

We quote what we said about this plan at the second city council hearing, at which the contest results were presented:

"Let me give a few examples of the problems that result from using this compactness measure. The district furthest out in the far Northeast stretches southwest, north of Roosevelt Boulevard in our solution to avoid zigzags, instead of cutting across the northeast tip of the city in a relatively straight line. The north–south slicing of Center City and South Philadelphia splits coherent neighborhoods. Not using Roosevelt Boulevard as a boundary combines disparate neighborhoods in Districts 1, 2, 3, and 8. Note that the 8th
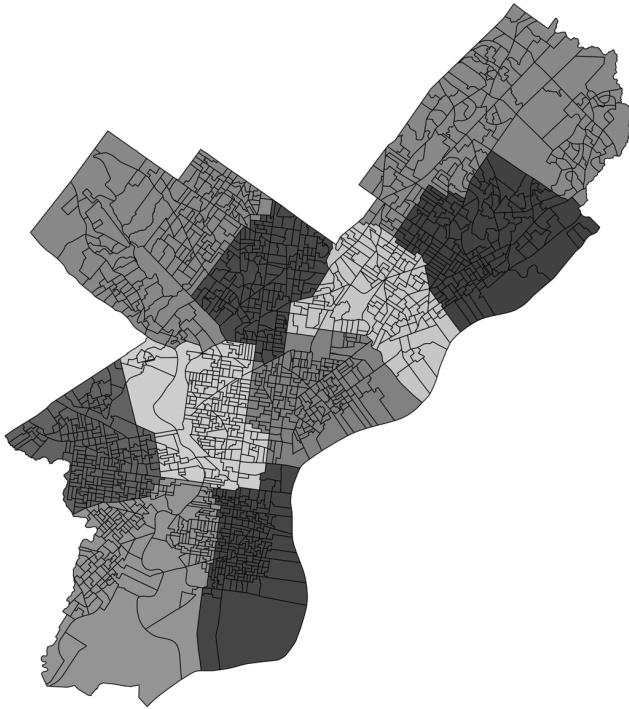
Figure 5: Our winning plan for compactness has a high compactness score, but does not respect neighborhood identities.

Figure 6: This map shows that districts can be adjusted to better match neighborhoods in the southern part of the city.

ward is divided into three pieces. Two other wards are split three ways as well. Nevertheless, starting from this solution and adding coherence by putting together neighborhoods would produce a plan that keeps the districts reasonably "compact."   (Murphy 2011)

We then repeated our offer to generate (for free) multiple plans keeping wards intact, which again was not accepted.

Because we could not advocate this plan, we offered a second one in the testimony to show how one could better respect neighborhoods by shifting the boundaries in the two districts at the southern end of the city. Shifting the upper part of the boundary of the lower right district to the left to a river and the lower portion to the main street north–south street (Figure 6) reduced the income spread in the affected districts on the left by a substantial amount.

What The City Council Did

The city council recognized that the districts had to be improved and constructed a plan that fixed the most egregious aspects of Districts 5 and 7. The map
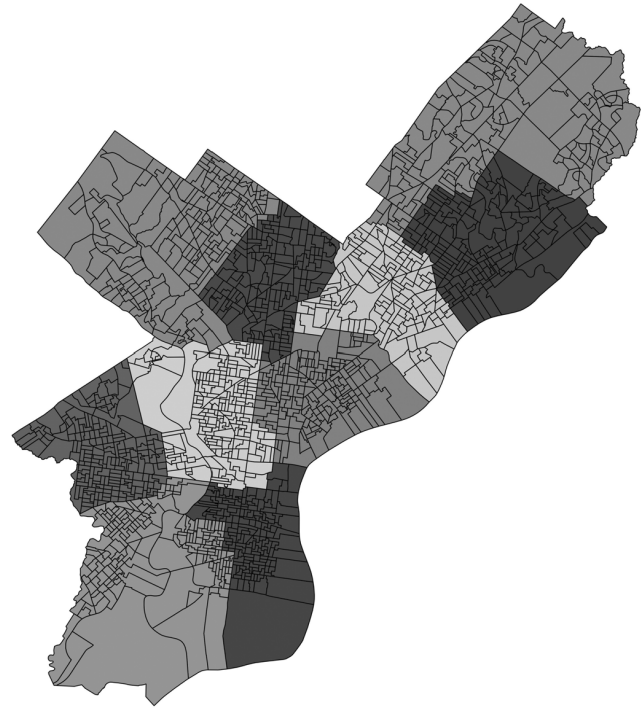
it finally chose (see Figure 7) is based on the proposal of two members of the city council.

Although no documented evidence exists that the members of the city council committee designing the districts used anything from the contest, the contest and pressure from public interest groups probably influenced their design to be more compact. Although the map looks much better than the previous districts, the boundaries, nevertheless, were clearly adjusted to help the incumbents. For example, the incumbent in District 5 extended his district further south to enhance his fundraising opportunities.

District 5 has a tail on the right to include the councilperson's home, while retaining enough of North Philadelphia, the city's poorest section, to give the incumbent an easy victory. A key feature of the new districts is that the middle-class and professional neighborhoods have been distributed among different districts to lessen their clout, but provide fundraising opportunities for the incumbents. In a few districts, this means the incumbent could be at risk in several years, given the rate of expansion of
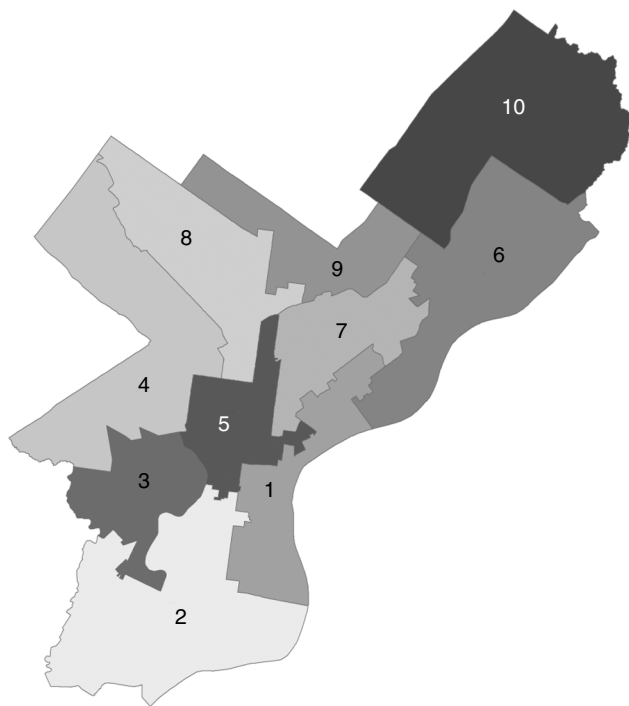
Figure 7: The city council selected this map.

the neighborhoods that are populated by middle-class professionals.

Conclusions

We won the contest in the compactness category and generated more (fully valid) solutions that keep wards intact than all other contestants combined, while acting simply as citizens. We most likely helped a councilwoman get a better district with one of our genetic algorithm-generated solutions and made clear the power of OR to a large audience. On the technical front, we developed an aggregation approach for producing much larger districting plans than in the past and used those solutions to generate multiple good solutions. Furthermore, we got excited and had a lot of fun with our zero-budget project.

The lessons learned in this project are in two areas—using analytical methods to solve real problems and understanding the deeper issues in designing districts. The key to our success was to not confine ourselves to the OR canon and to do whatever it took to reach our goal. This meant that we did not treat solving a standard optimization as the end point. Instead, when we could not get the data in a form we needed, we used the optimization to find a good starting point for using heuristics that we developed as we played with alternative maps. Next, we recognized that we did not have to choose between classical optimization and genetic algorithms but should use both, classical optimization to start us off and genetic algorithms to explore around the optimization solution. That is, we adapted our methods to solve the problem.

Being clever is not enough. Once one goes beyond the canon, one should then assess what else to add. Our project raises at least two questions: First, given all the alternative objective functions in Young (1988), can one of them be used as a proxy that evaluates the relative merits of districting plans? More generally, when do proxy objective functions give meaningful solutions? We investigated the relationship between compactness measures and peoples' perceptions of the goodness of districts in Chou et al. (2013). The next question is more general in IP. To what extent can genetic algorithms provide multiple solutions around the solution to an integer program and provide a principled approach to sensitivity analysis of integer solutions? We have begun work on that.

We found that treating the districting problem as a straightforward optimization problem is not entirely satisfactory. We learned from this exercise that gerrymandering is like pornography—you know it when you see it. However, defining what constitutes well-designed districts is hard, because good districts have so many dimensions, and different population groups and politicians have different definitions of good. To fix obsessively on one measurable criterion, such as minimizing population differences, or on one of the definitions of compactness is a mistake.

The problem is that redistricting is a game with multiple players who have distinct interests. It is now common knowledge that political parties use redistricting for advantage, with each party using redistricting to increase the numbers of its members in elected office. Jockeying between parties was not the issue in Philadelphia because the city is dominated by one party. Redistricting here is about enhancing or diminishing the power of individual members of the city council. Elected officials play against each other

to both make their districts safe for reelection and to capture enough wealthy residents so that they can raise the funds that buy influence over other elected officials. Civic-minded groups want a fair outcome that gives citizens a voice in government.

Essentially, we played a part in the game on the side of public interest groups that were pushing for improved districts; we also garnered recognition for OR technologies. The city council's redistricting committee originally wanted to keep the long tails extending into the northeast part of the city to dilute to power of an aggressive ward leader. However, the added hearings, citizen involvement, and less-gerrymandered alternatives presented through the contest gave the councilwoman in District 7 the opportunity to eliminate the tails. An activist who supports her in her district came to us and asked for advice; we gave him all the maps we generated, saying he could use them as alternatives.

Rather than an optimization, a process is needed to balance the interests of the players, recognizing that politicians have legitimate interests, while focusing on the ideal of representative democracy. A first step in the process should be to define legitimate objectives, even if they are surrogates for the real objective, and agreed-upon constraints. The second step is to develop a multiplicity of solutions to form the basis of the discussions. (This is essential for participation by the general public, which does not have direct access to districting data and tools.) The basic ground rules for good districts, such as making every effort to keep wards or neighborhoods intact, should be established and have public agreement well before the redistricting process begins. Analytic techniques are useful for generating good starting solutions. With reasonable constraints on the redistricting plans, such as honoring neighborhoods or political subdivisions, using analytic methods to generate a large but not overwhelming number of choices quickly and easily is possible.

Next, the process of developing the new districts should involve all interested people, and the redistricting choices should be constrained to focus on principled, broadly acceptable plans. Creative use of computational methods can, as we demonstrated, present all stakeholders with a substantial pool of such plans. Ultimately, the final districting will and should be done by people who have a stake in and understand the nuances of the neighborhoods and the city.

Although measures of compactness can point out serious gerrymandering, the best safeguard against egregiously gerrymandered districts is an open discussion of plans, placing those plans on the Web for citizens and public interest groups to comment. Citizens should be able to add plans and comment on existing proposals. Refinements of those plans, along with commentaries explaining why the refinements were made, should be posted periodically, again providing for open dialogue and modifications by citizens. A small set of final plans should be posted with a comment period before the final vote.

A process that demonstrates intent to be fair is just as important as the final redistricting, because some neighborhoods and interest groups will almost certainly not receive all the gains they seek.

Appendix A. The Integer Programming Formulation and Heuristic Solution Procedure for the Districting Problem

Assume that a geographical zone to be districted (e.g., a city or a state) consists of a set I of mutually exclusive, collectively exhaustive population units, with P_i representing the population in unit i. We assume that the district centers are to be selected from a set J of candidate centers. The sets I and J can be the same population units, but do not have to be. Let $d(i, j)$ represent the distance from i to j, measured according to some suitable metric, such as squared Euclidean distance from i to j. Each of the K districts needs to contain a population between POPLOW and POPHIGH. Decision variable $X(i, j)$ is 1 if population unit i is assigned to the district center at j (and 0 otherwise). Decision variable $Y(j)$ is 1 if the district center at j is selected to be a center for a district (and 0 otherwise). The standard p-median (or facility location) approach to the districting problem (DP) follows.

Problem DP:

Minimize $\sum_i \sum_j d(i, j) X(i, j)$

s.t. $\sum_j X(i, j) = 1$ for all $i \in I$, (A1)

$\text{POPLOW} \leq \sum_i P_i X(i, j) \leq \text{POPHIGH}$
for all $j \in J$, (A2)

$X(i, j) \leq Y(j)$ for all $i \in I$ and $j \in J$, (A3)

$$\sum_j Y(j) = K, \tag{A4}$$

$$X(i, j),\, Y(j) \text{ are binary decision variables,} \tag{A5}$$

$$\text{Set of contiguity constraints.} \tag{A6}$$

Note that constraints (A1)–(A5) alone represent the standard *p*-median model; solving DP without constraint (A6) can result in districts that are not entirely contiguous. However, given the nature of the objective function coefficient $d(i, j)$, we expect the solution of DP to be almost contiguous. We discuss the exact implementation of the contiguity constraints after we present our heuristic solution procedure.

In our solution procedure, we considered three sets of candidate centers J:

$J_1 =$ set of 1,687 ward divisions;

$J_2 =$ set of 66 wards, with each ward containing a number of ward divisions (subsets) from set J_1; and

$J_3 =$ set of 155 neighborhoods.

Sets J_1, J_2, and J_3 represent varying levels of population aggregation in the city and the populations of the elements of sets J_1, J_2, and J_3 add up to a population of about 1.5 million, the population of Philadelphia. Moreover, if districting is to be accomplished at the ward level with set I in DP the same as set J_2, we have a small integer program that can be solved to optimality using available solvers, such as CPLEX/GAMS. We therefore consider the solution of the 1,687 population unit (ward division) problem in the three-step heuristic procedure stated next (i.e., $I = J_1$ and $J = J_1$). Because the heuristic procedure works by first aggregating and then disaggregating population units, we refer to our heuristic procedure as the *d*ata-*a*ggregation-*d*is-*a*ggregation heuristic or the DADA heuristic.

DADA Heuristic:

Step 1. Solve problem DP with set $I = J_1$ and set $J = J_3$, the set of neighborhoods, *using only constraints* (A1)–(A5), to obtain a set $J_3(K)$ of K candidate centers. Set $J_3(K)$ is a subset of set J_3. Note that the districting solution obtained in this step may not be contiguous.

Step 2. Identify a subset $J_1(K)$ of J_1, where the elements of J_1 are closest to any element of $J_3(K)$ obtained in Step 1 using squared distance. Fix the members of $J_1(K)$ as the centers for Step 3. (Because $J_1(K)$ need not be the optimal set of district centers, procedure DADA is only a heuristic.)

Step 3. Solve DP with $I = J_1$ and $J = J_1(K)$ and all constraints (1)–(6) that include the contiguity constraints.

Because $|J_1(K)| = K$ and the centers for the districts are now fixed, we can add the preprocessing step $X(i, i) = 1$ if i is a member of $J_1(K)$. Once DP is solved in Step 3 with the contiguity constraints, we obtain a contiguous district solution that satisfies the population bounds POPLOW and POPHIGH. However, this is still a heuristic solution because centers are fixed in Step 2. The objective of the genetic algorithm described in Appendix B is to improve the solution from Step 3 by performing a local search.

The intuition behind the contiguity constraints is that each center belongs to a different district and can serve as a root node for a subtree of population units that are in the same district. The constraints enforce a flow from each population unit in a district to the root node in that district through contiguous population units in the same district. By requiring the outflow from a population unit that is not a root node to be one unit greater than the inflow, the constraints ensure that the only node with a net inflow is the root node, which enforces contiguity in the district. Shirabe (2009, p. 1059) provides details.

Let $k = 1, \ldots, K$ and let variable $z(i_1, i_2, k)$ represent the total flow from node (population unit) i_1 to node i_2 when both i_1 and i_2 are assigned to district k. Let A be the set of pairs of population units that are adjacent: $A = \{(i_1, i_2) \mid i_1$ and i_2 in set I and adjacent$\}$. Note that if the ordered pair (i_1, i_2) is a member of A, (i_2, i_1) is also a member of A. Let M be a large number chosen to be an upper bound on the number of population units to be contained within any one district (a classic big M).

To prevent flows between two districts, the following constraint permits an inflow to node i_1 only if node i is assigned to district k.

$$\sum_{(i,\, i_1) \in A} z(i, i_1, k) \leq M X(i, k)$$
$$\text{for all } k \in K \text{ and } i \notin J_1(K) \tag{A7}$$

The next constraint forces a net outflow of one from every node that is not a root node if both nodes are in k.

$$\sum_{(i_1,\, i) \in A} z(i_1, i, k) - \sum_{(i,\, i_1) \in A} z(i, i_1, k) = X(i, k)$$
$$\text{for all } k \in K \text{ and } i \notin J_1(K). \tag{A8}$$

Note that the outflow $\sum_{(i_1,\, i) \in A} z(i_1, i, k) = 0$ when i_1 is not in District k. Thus, the combination of Equations (A7) and (A8) ensures that i and i_1 are both in District k when flows are positive. The last constraint caps the flow out of district centers to avoid unbounded solutions by imposing a bound on the maximal flow collected by each district's root node from all other nodes that belong to the same district.

$$\sum_{(i,\, i_1) \in A} z(i, i_1, k) \leq M - 1$$
$$\text{for all } k \in K \text{ and } i \in J_1(K). \tag{A9}$$

Appendix B. The Evolutionary Algorithm

Our evolutionary algorithm (EA) was written in MATLAB and centers around the script main.m as the main program. The algorithm does not use recombination (crossover); it relies principally on a special form of mutation, described later, to introduce variance. We eschewed recombination because of time constraints and because it tends to produce noncontiguous solutions. Perhaps with much longer-running explorations, recombination would prove useful. We note that although genetic algorithms ordinarily use some form of recombination, historically the evolutionary programming (EP) community has not. Our algorithm is

best classified as a form of EP, within the umbrella class of EAs.

At the outset, we attempted to evolve contiguous solutions, but were unsuccessful because we were working under a tight time constraint. Consequently, as we describe next, we obtained four contiguous solutions as starters and evolved new contiguous solutions from them.

The main.m program first initializes the run and then calls another procedure to run the main loop of the algorithm. We now describe these two aspects separately.

Initialization

In main.m, entry of data leads to instantiation of the following variables:

wardPops: An array (or table) that maps ward number (1–66) to its population value in the 2010 census.

wardCoords: An array that maps the ward number to both X- and Y-coordinate values. The resulting point is the geographic center of the ward.

wardAdjacent: A 66×66 adjacency matrix, a symmetric table of 1s and 0s, with 1 indicating that the wards of the corresponding row and column are immediately adjacent. With 1s on the diagonal, wards were treated as immediately adjacent to themselves.

distanceMatrix: Using the X- and Y-coordinate data for the wards, we generated an interward distance data file offline that was loaded during program initialization into this array.

The program then operates with these four basic sources of data, plus the following parameter initializations.

numDistricts: The number of districts to be designed (set to 10).

targetPop: The total population divided by the number of districts (numDistricts).

slackAllowed: Set to 0.025. We used this to constrain the district populations in a solution so that they were individually all within 2.5 percent of the target population.

theMutationRateHigh, theMutationRateLow: Set to 0.15 and 0.01, respectively. Innovatively, or at least unconventionally, we used both mutation rates during each run, as we explain later. This served to reduce the time used to explore the parameter space and tune the algorithm.

lambda: Set to 3. We used $2 * lambda$ as the number of mutated solutions each member of the population produces in each generation. Therefore, in each generation, each member of the current population spawns six daughter solutions by mutation.

numGenerations: Set to 2,000. We halted each run after 2,000 generations; we arrived at 2,000 using a modest tuning effort.

thePop: The population. We entered the initial population for each series of runs at start up. We used four different initial populations, each consisting of 50 contiguous solutions to the problem. Each of these initial populations was generated from one of four distinct single-contiguous solutions, each of which was produced by other team members using the integer program. In each case, from a single starting solution, we generated 50 new contiguous solutions using the mutation procedures used in the main evolution loop. Here, we simply ran the procedure on the original and any found contiguous solutions until we obtained the required number of solutions. As it happened, none of the solutions in the four populations were feasible with respect to population deviation.

Main Loop

The main loop of the algorithm is essentially simple. For each generation (i.e., 2,000 times), the algorithm performs the following three actions.

1. Produces $2 * lambda$ daughters for each member of the current population using what we call neighborhood mutation (i.e., only wards with neighbors assigned to another district are mutated), with the permitted mutation values limited to the current value (assigned district) of the ward and the neighboring values. In addition, each solution produces lambda daughters using the high mutation rate and lambda daughters using the low mutation rate.

2. Assesses the fitness of each member of the augmented population. The objective function value is the maximum within-district distance between wards for a given district assignment (solution). We seek to minimize this. Solutions that are noncontiguous are heavily penalized to ensure that they do not appear in the next generation. Solutions that violate the population-size constraint are penalized less severely, but in proportion to the size of the violation. Therefore, fitness is the objective function value plus any penalties, and we seek to minimize it.

3. Selects the next-generation population by culling the best popSize solutions from the augmented population. We scored the solutions on fitness, as described previously, by computing the maximum within-district distance of a solution, plus penalties for violations of contiguity and population size constraints. We then ranked the solutions in the population and produced the next generation by taking the best popSize of these solutions. This use of ranking selection, along with not having recombination, is typical of EP algorithms.

Control

A control procedure (called go.m) directs the main program and its main loop. It basically sets the number of runs (each run consisting of 2,000 generations for a single starting population), sets which starting population will be used (we had four in all), and collects across all the runs and each generation any feasible new solution encountered. We found 116 of these.

References

Azavea (2006) The gerrymandering index. White paper. Accessed June 13, 2013, http://cdn.azavea.com/com.redistrictingthe nation/pdfs/Gerrymandering_Index_Whitepaper.pdf.

Bozkaya B, Erkut E, Laporte G (2003) A tabu search heuristic and adaptive memory procedure for political districting. *Eur. J. Oper. Res.* 144(1):12–26.

Bozkaya B, Erkut E, Haight D, Laporte G (2011) Designing new electoral districts for the city of Edmonton. *Interfaces* 41(6):534–547.

Chou C, Kimbrough SO, Murphy FH, Sullivan-Fedock J, Woodard CJ (2013) On empirical validation of compactness measures for electoral redistricting and its significance for application of models in the social sciences. *Social Sci. Comput. Rev.*, ePub ahead of print April 9, http://ssc.sagepub.com/content/early/2013/04/09/0894439313484262.

Garfinkel RS, Nemhauser GL (1970) Optimal political districting by implicit enumeration techniques. *Management Sci.* 16(8):B495–B508.

Hess SW, Weaver JB, Siegfeldt HJ, Whelan JN, Zitlau PA (1965) Nonpartisan political redistricting by computer. *Oper. Res.* 13(6):998–1006.

Mehrotra A, Johnson EL, Nemhauser GL (1998) An optimization based heuristic for political districting. *Management Sci.* 44(8):1100–1114.

Murphy FH (2011) Testimony by Fredric H. Murphy. Accessed June 1, 2013, http://opimstar.wharton.upenn.edu/~sok/phillydistricts/doc/CompactnessTestimony.pdf.

Murphy FH, Hess SW, Wong-Martinez CG (2013) Politics. Gass S, Fu M, eds. *Encyclopedia of Operations Research* (Springer, Berlin).

Shirabe T (2009) District modeling with contiguity constraints. *Environ. Planning B: Planning Design* 36(5):1053–1066.

Williams JC (2002) A zero-one programming model for contiguous land acquisition. *Geographical Anal.* 34(4):330–349.

Young HP (1988) Measuring the compactness of legislative districts. *Legislative Stud. Quart.* 13(1):105–115.

Zoltners AA, Sinha P (1983) Sales territory alignment: A review and model. *Management Sci.* 29(11):1237–1256.

Ram Gopalan received his PhD from the Operations Research Center at MIT. His research focuses on transportation science, logistics, and public policy models. He is on the faculty at Rutgers University, Camden.

Steven O. Kimbrough is professor of operations and information management at the University of Pennsylvania, where he has been on the standing faculty since 1984. He works as a decision scientist, with an overarching interest in rationality and computation. His principal research areas are in procedural (post-classical) game theory, evolutionary computation, and applied logic. He also works in the area of text mining and analytics. He is an inventor on three related patents in this area. His research features extensive use of both computational methods and formal logic. He is author of the book *Agents, Games, and Evolution: Strategies at Work and Play* (2012) and he is actively at work in exploring individual learning algorithms in the context of strategic decision making. He was instrumental in founding the business analytics concentration at The Wharton School.

Frederic H. Murphy is a professor emeritus in the Fox School of Business, Temple University. He works mainly in the areas of modeling economic equilibria both with classical methods and agent-based techniques, energy-market forecasting, and energy policy. He has been involved in studying local economic policy issues, including advising the Tax Reform Commission of the City of Philadelphia and analyzing the economic impact of local casinos. He is currently advising the King Abdullah Petroleum Studies and Research Center in Riyadh, Saudi Arabia on building energy/economy models of Saudi Arabia. His publications range from policy analyses to algorithms for computing economic equilibria and solving optimization models.

Nicholas Quintus joined the redistricting team while pursuing a master's degree in geography and urban studies at Temple University in Philadelphia. Nicholas was attracted to the project because it combined the subjects of public policy and geography. He is employed as a geographic information systems specialist in the Philadelphia Water Department's Office of Watersheds. As a lifelong voter and citizen of the City of Philadelphia, Nicholas intends to remain focused on improving the redistricting process at all political levels.