

Portland Public Schools

Better school district boundaries using Gurobi

The Challenge of Redistricting

In a dynamic city, the population of school-age children is always in flux. Unfortunately, the set of buildings available for use as public schools is not nearly as fluid. As a result, it is a constant challenge to choose appropriate assignments of school children to schools in order to achieve desired goals like minimizing travel distance, allowing kids who live near each other to attend the same schools, and achieving desired minimum and maximum class sizes.

This "redistricting" problem is challenging, owing to the large number of decisions that are potentially in play:

- Each school building can be used as either an elementary school, a middle school, or a K-8 school.
- Kids from each city block can potentially be assigned to one of several different nearby schools. A typical city will have dozens of school buildings and thousands of distinct city blocks, leading to an astronomical number of potential options.
- Within this context, parents of young children can be quite discerning customers, so the resulting decisions typically receive a tremendous amount of scrutiny.
- Districting choices must be both fair and defensible.

Traditionally such decisions have been made by hand, requiring many hours of tedious manual work and lots of hand-wringing over whether the resulting assignments were really as good as they could be. Employees of Portland Public Schools have recently built an optimization model that solves the redistricting problem as a Mixed-Integer Programming (MIP) problem using Gurobi Optimizer. This MIP model yielded better results in less time.

The Optimization Model

After being involved in many iterations of the redistricting process, using tools ranging from pencil-and-paper analysis to computer programs that implemented basic redistricting heuristics, Shawn Helm, a long-time Portland Public Schools employee, decided that there had to be a better way. After reading academic papers by Hayri Önal on using MIP to redistrict political boundaries, Shawn saw the potential for using similar methods in school boundary redistricting.

Using a number of online resources, including the Gurobi modeling and interactive examples as well as relevant academic papers ("A Model of Contiguity for Spatial Unit Allocation", by T. Shirabe), Shawn together with Sahan Dissanayake, assistant professor of Economics at Colby College and Will Kearney, data analyst at Portland Public Schools, began building a MIP model to help solve the redistricting problem in Portland. Their initial model led to the launch of an official project, which eventually resulted in a custom Python application that can propose new district boundaries in a few hours.

Objectives:

- Travel time: Kids should go to the nearest school.
- Contiguity: Kids should go to the same school as their neighbors.
- Target enrollment: Desired school size.
- Continuity: Kids should go to the same school that they attended before redistricting.

Data:

- The school districts were divided into geographic planning areas. The location of the planning area was provided, as well as the number of kids from each grade that lived there, the schools that they attend, and the neighborhood demographics.
- Location, capacity, and which grades (K-8, K-5, 6-8, closed, etc.) the school serves were also provided.

Decision variables:

- Determine for each planning area and grade, which schools kids should attend.
- Determine for each school, which grades (K-8, K-5, 6-8, etc.) the school should serve.
- Determine if any currently closed school should be re-opened.

Constraints:

- Each school has a capacity.
- Each grade at each school must have at least two classes of kids.
- Each elementary school feeds one middle school.
- It is desirable that districts are contiguous.

The approach is a heuristic that follows a multi-phase strategy:

- Phase 1: Ignore contiguity constraints, choose grade range (K-5, K-8, etc.).
- Phase 2: Fix grade ranges and feeder patterns and add contiguity soft constraints.
- Phase 3: Minimize the number of disconnected regions.
- Phase 4: Fix the number of regions, maximize adjacencies (to smooth boundaries).

Portland Public School District

Better school district boundaries using Gurobi

The Solution

The solution tool encompassed an SQL-Server database that stored school locations, geographic planning areas, population data - number of kids in each planning area for each grade, and optimization scenarios.

An optimization scenario had different parameters to force “this” school to be K-5, reopen “this” closed school, etc. In addition, an optimization scenario enables long-term planning, based on predicted population in 10 years, for example.

The solution tool had a Graphical Information System (ArcGIS) that allowed one to visualize data and results and pull data from the SQL-Server.

Last, the solution was built using the Gurobi Python API, which pulled data from the SQL-server to create variables and constraints.

This decision support tool was used to help make decisions. The tool enabled multi-objective optimization of various conflicting objectives, such as minimizing travel distance, achieving smooth boundaries, creating uniform class sizes, and maintaining continuity with previous years. Also, the decision support tool enabled short-term planning using today’s data and long-term planning using predicted future data.



Figure 1: This is a solution generated by the tool, where contiguity constraints and adjustments are considered in the MIP model. Each color represents a different school district.



Rock Creek Middle School in Happy Valley

About Portland Public Schools

Portland Public Schools, founded in 1851, is a PK-12 urban school district in Portland, Oregon. With more than 49,000 students in 78 schools, it is one of the largest school districts in the Pacific Northwest.

Why Gurobi?

The team at Portland Public Schools chose Gurobi for a number of reasons:

- The Gurobi Academic Program allowed them to use a state-of-the-art solver for their project without a large expense.
- The model was large and complex enough that the project would likely have failed without a high-performance solver.
- Gurobi Online Examples taught them quite a lot about best practices in optimization modeling.
- The Gurobi Python Interface made it easy to build and maintain their model.