Python Multiwinner Package Manual

December 5, 2016

1 Introduction

This is a rudimentary manual for using the Python Multiwinner Package software. The package is implemented in Python (2.7.3) and provides a number of features:

1. generation of preference profiles based on the two-dimensional Euclidean domain;
2. computing winners of multiwinner elections;
3. visualizing election results;
4. running series of experiments.

2 Core Components

Below we describe the core components of the package. The main program here is `experiment.py` which takes care of running a single experiment (generating a 2D Euclidean election, computing results according to various rules, and preparing visual representation of the results). This program uses `2d2pref.py`, `winner.py`, and `visualize.py` to achieve its goals.

2.1 Running an Experiment

Program `experiment.py` can be ran as follows:

```
python experiment.py <description.input
```

where `description.input` is a file specifying how the experiment should be conducted (what 2D Euclidean election to generate, which rules to run, and for what committee sizes). Below we provide an example of an `.input` file that uses most of the features of `experiment.py`:

```
candidates              # switch to generating candidates
gauss 1 0 0.5 125       # generate 125 points with sigma=0.5,
                        # centered at (1,0)
```

```
uniform -2 -2 1 1 50     # generate 50 points distributed uniformly
                         # on the square (-2,-2)(1,1)
voters                   # switch to generating voters
circle 0 0 2 400         # generate 400 points distributed uniformly
                         # on a disc centered at (0,0) with radius 2
generate input-data      # save the generated points to file input-data.in
                         # generate preference-based election and save
                         # it to file input-data.out
stv  20 input-a-stv      # compute the election result for 20 winners using
                         # STV and save the result to input-a-stv.{win,png}
stv  40 input-b-stv      # compute the election result for 40 winners using
                         # STV and save the result to input-b-stv.{win,png}
```

There two main outcomes of running `experiment.py`. The `.win` files which contain the sets of candidates and voters (both including their positions in the 2D model of the generated election) and the winner set (also including their 2D positions). The `.png` files contain visualizations of the election results. In most typical ways of using `experiment.py`, a single run generates only a single election, but possibly runs several rules on it (possibly for several different committee sizes).

2.2 Converting 2D Data to Preference Orders

2.3 Computing Elections Winners

2.4 Visualizing Elections

2.5 Additional Tools