Problem Statement - ClinicFlow

Team:
Maxim Vasiliev – 400043983
Susie Yu - 000955758
Karl Knopf - 001437217
Weilin Hu - 001150873
Yunfeng Li - 001335650

The Josef Brant Hospital pre-operative clinic (the client) schedules upwards of 50 patients per day. The appointment times are digitized, yet manually chosen by clinic staff. Once at the clinic, each patient undergoes a varying set of procedures with different durations. While staff have a good feeling of how to schedule patients, mistakes and inefficiencies often occur considering the numerous constraints and temporal variation of events. The client has approached us to explore the potential of simulating the clinic and analyzing the results. This would help the staff foresee potential problems, and allow them to test how any additions to the clinic would affect their bottom line.

We propose a tool which allows hospital staff to simulate patient flow through the clinic. The user would input appointment data and constraints (E.g. employee work hours, procedure duration, procedure order), and receive as output various metrics and visualizations resulting from the simulation of the clinic. The user would then be able to modify some constraint, re-run the simulation, and compare the effects of the modification. This would assist staff in controlling daily patient flow and deciding on whether to enact proposed changes.

Features

Quick and easy UI. Clinic staff are short on time; this tool should only help them.

Customizable. Staff are able to add constraints. Tool can be applied to other clinics or similar environments.

Accessible. Device agnostic and accessible by all staff.

Persistent. Resulting data can be referred back to and compared.

Challenges

Modeling the clinic, including establishing clinic routine (lunch breaks), scheduling rules (blood test always before x-ray), determining event time variability (patient booked vs. actual arrival, procedure duration), quantifying flexibility (Interview duration can be squeezed, electrocardiogram cannot), and encoding the math and logic required to compute and simulate patient flow through the clinic.

Designing the interface, including the method by which the user inputs the schedule data or constraint changes, and how the system formats and delivers the results. The input should be intuitive and customizable (adding a new procedure with no coding), and the output informative (graphically present

the client flow so that bottlenecks can be visualized). This also concerns the way result data will be stored; ideally simulation results could be referred back to.

Other challenges might include:

Considering patient appointments across multiple days, and suggesting changes to scheduling.

Allowing the model to learn and correct itself based on past performance metrics.

Assumptions

- That the patients and the health care workers do not have unscheduled breaks during the simulation of the clinic

- That we will be able to accurately capture all of the variance in the clinic with a model

Objectives

Our first objective is to model the clinic:
- Interview the client to capture all features of the system.
- Clean and analyze data to determine the variability in each component. We will be provided data on available procedures and past patient flow through them, including appointment time, arrival time, and start and end times for each procedure the patient undergoes.
- Model each component as a distribution.

From there we will build the simulation engine:
- Deciding on the deliverable form of the application. We have opted to use python, due to simulation package availability and ease of integration into a web application.
- Researching and incorporating math and system constraints, including queuing, Markov chains, and stochastic processes.
- Testing the accuracy of the resultant computations.
- Formatting and outputting the results.

The back end:

- Learn database frameworks, and select the best one for the project
- Integrate the database management software with the simulation

The front end:
- Set up the server and build the user interface.
- Integrate user inputs with the back end simulation algorithm.

Other:
- Documentation, guides, reports.

References

1. Rosocha L, Vernerova S, Verner R. (2014). MEDICAL STAFF SCHEDULING USING SIMULATED ANNEALING. Quality Innovation Prosperity. doi: 10.12776/QIP.V19I1.405
2. Leksakul K, Phetsawat S. (2014). Nurse Scheduling Using Genetic Algorithm. Mathematical Problems in Engineering. doi:10.1155/2014/246543