What Can we Learn

This model should highlight and offer insight into how to best plan the order of patient rounds done by attending physicians. As mentioned above, it is critical to be able to release patients as safely and time efficient as possible. This both reduces cost and improves patient satisfaction. With this model, we hope to learn more about the patient release process and to identify and optimize key variables in reducing time to release.

More succinctly the guiding question is as follows: How might we reduce patient release times in the MICU by thoughtfully arranging the order in which attending physicians visit their patients. For example, we may find that attending physicians should visit patients that have been highlighted for release first. And that this operational strategy is positively or negatively related to patient release time. These insights would then guide procedural recommendations the MICU.

I plan to conclude which strategy for rounds is most effective. In addition, I plan to explore how one might decouple release from regular patient rounds and how this might affect process variability. For example, given the scenario that all patients – not just those looking to be released– will be visited, how does stratification of the type of visit (health or release) affect overall patient release time.

I plan to initially set up the model in general fashion so the results may be externally valid to other ICU or patient release operations.

A Guide to Implementation

Who are the Agents(Behaviors | Properties):

1. Attending Physicians (Speed of Round, Conservative or liberal?, | release-patient, pick next room)
2. Patients (Wait time, release-type, acuity, length of stay | Release-Me!)
3. Patient Transporters (time-to-ready, | transport
4. Possibly fellows, residents and interns but I do not think they will be necessary as I have not observed them having unique characterisitics (Positive or negative effect on round speed)
5. Medical Floor Physician (Record Review Speed, Open Bed? | Admit patient)
6. Long Term Care Facilities (Open Bed? | Admit patient)

System Parameters

1. Number of patients or patients per day inflow
2. Cost per day (whole days 1,2,3 etc.)
3. Patient Acuity
4. Number of patient releasers
5. Number of beds

Time-Step (Day)

1. Attending visits patient
2. Attending assesses patient related to acuity

Beyond its value in facilitating communication, the computational nature and ease of extendability will allow others to build off the model to understand tangential phenomena. This also means that the model can be improved by integrating real hospital data into the model.

Similarly, the use of machine learning methods and GIS may become important when trying to optimize the system. Machine learning may be important to select a strategy that is best suited to reduce release times. GIS could be used to help define spatial constraints in the model. For example, it may be possible to upload a floor plan of the ICU of interest and use image recognition to define spatial constraints.

Agent Based Modeling of this phenomena has several distinct advantages over alternative simulation methods. First, many model paradigms require extensive data collection and can take years to process the health data in a way suitable for simulation. In addition, health data is notorious for being difficult and lengthy to process. Second, other modeling paradigms like monte carlo simulation or discrete event optimization do not allow interested stake holders to gain insight from the model construction itself. Specifically, interested stakeholders have to rely more on the modelers to have correct assumptions. With ABM, the assumptions are clear and up for debate.

The current model to analyze the operational effectiveness of patient release is often judgement based that depends on expert judgement. While timely, expert judgement often comes feeling and experience rather than a stated assumption. To me, it appears exceedingly clear that Agent Based Modeling will aid in the re-engineering of the patient release workflow.