Block Scheduling for a Pediatric Residency Program

Peter Mayoros
University of Michigan
November 10, 2014
My Collaborators

• Research Team:
  – Amy Cohn, Ph.D.
  – Zak Verschure
  – Young-Chae Hong
  – Ji Wang
  – William Pozehl

• Contacts at the UM Health System:
  – Edward O’Brien, M.D.
  – Jenny Shin, M.D.
Presentation Outline

• Background
• Motivation
• Model Formulation
• Model Implementation
• Results
• Future Work
Pediatrics at Michigan

• Based in C.S. Mott Children’s Hospital
• Around 100 residents at Mott every year
• The hospital is very diverse in the programs it offers
Services in Pediatrics

• There are many different areas in Mott that residents must gain experience in:
  – E.g. Neonatal Intensive Care Unit (NICU), General Care, Hematology & Oncology (HemOnc)

• Every month each of these services must be adequately staffed and there is a minimum and maximum number of residents allowed in each service
Residency at Michigan

- After four years in medical school the students are “matched” to a residency program
- These first year residents are called “interns”
- After the first year they become “senior residents”
- Every resident has a minimum and maximum amount of time (in half months) that they must spend in each service
- We were tasked with compiling a block schedule for just the interns
Service Pairs

• A cuplet of services that can be worked during the same month.

• The first service in the pair is worked for the first half of the month and the second service is worked for the second half of the month.

• Different examples of service pairs could be:
  – General/general or NICU/general

• Different combinations can also be determined as hard or not
### What is a Block Schedule?

Assigning service pairs to residents over the course of a year

**NOTE:** vacation as a service

<table>
<thead>
<tr>
<th></th>
<th>Service Pair 1</th>
<th>Service Pair 2</th>
<th>Service Pair 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe Smith</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>July</td>
<td>First Half</td>
<td>Second Half</td>
<td>First Half</td>
</tr>
<tr>
<td></td>
<td>General</td>
<td>General</td>
<td>NICU</td>
</tr>
<tr>
<td>August</td>
<td>First Half</td>
<td>Second Half</td>
<td>First Half</td>
</tr>
<tr>
<td></td>
<td>NICU</td>
<td>HemOnc</td>
<td>General</td>
</tr>
<tr>
<td>September</td>
<td>First Half</td>
<td>Second Half</td>
<td>Second Half</td>
</tr>
<tr>
<td></td>
<td>General</td>
<td>Vacation</td>
<td></td>
</tr>
<tr>
<td>Hard</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Service Pair 4</th>
<th>Service Pair 1</th>
<th>Service Pair 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>July</td>
<td>First Half</td>
<td>Second Half</td>
<td>First Half</td>
</tr>
<tr>
<td></td>
<td>HemOnc</td>
<td>NICU</td>
<td>General</td>
</tr>
<tr>
<td>August</td>
<td>First Half</td>
<td>Second Half</td>
<td>Second Half</td>
</tr>
<tr>
<td></td>
<td>General</td>
<td>General</td>
<td>NICU</td>
</tr>
<tr>
<td>September</td>
<td>First Half</td>
<td>Second Half</td>
<td>Second Half</td>
</tr>
<tr>
<td></td>
<td>HemOnc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Motivation

• Before:
  – Block Schedule was compiled by hand by the chief resident
    • This took a lot of time
    • If changes needed to happen after a draft was completed they had to rewrite the entire schedule

• Now:
  – A computer program compiles the schedule in less than 5 minutes
  – Changes are effortless
Sets:

\[ R : \text{ set of all residents that need to be scheduled} \]

\[ P : \text{ set of all service pairs} \]

\[ S : \text{ set of all services that need residents} \]

\[ M : \text{ set of months} \]

\[ D : \text{ set of desires. this is a list given to us by the chief resident that details the residents vacation requests} \]

\[ p_{si} \subset P : \text{ subset of service pairs } P \text{ that have service } S \text{ in half-month } i \text{ where } i \in \{0, 1\} \]

\[ h_p \subset P : \text{ set of pre-defined hard service pairs} \]

\[ C_d \subset D : \text{ set of candidate assignments for desire } D \]
Model Overview

Parameters:

$q_{ps}$: the number of half-months of service $s$ that are in service pair $p$, $\forall s \in S, p \in P$

$b_p$: if service pair $p$ is difficult 1, if easy 0, $\forall p \in P$

$u_{sm}$: an upper bound for the number of residents service $s$ needs in month $m$, $\forall s \in S, m \in M$

$l_{sm}$: a lower bound for the number of residents service $s$ needs in month $m$, $\forall s \in S, m \in M$

$u_{rs}$: an upper bound for the number of months required on service $s$ for the year, $\forall s \in S, r \in R$

$l_{rs}$: a lower bound for the number of months required on service $s$ for the year, $\forall s \in S, m \in M$

$l_d$: a lower bound for the number of desires filled for month, $\forall m \in M$

$u_d$: an upper bound for the number of desires filled for month, $\forall m \in M$
Model Overview

**Decision Variables:**

\[ X_{rpm} : \text{ Whether or not resident } r \text{ is scheduled to service pair } p \text{ on month } m \quad \forall r \in R, p \in P, m \in M \]

\[ Y_{rm} : \text{ Whether or not resident } r \text{ works } 3 \text{ or more hard services in a row beginning on month } m \quad \forall r \in R \]

\[ Z : \text{ Helper variable for maximum number of times a resident works } 3 \text{ difficult shifts in a row.} \]

**Objective Function:**

- Constant

**Variable Restrictions:**

\[ X_{rtm} \in \{0,1\} \]
\[ Y_{rm} \in \{0,1\} \]
\[ Z \in (R) \]
Each Resident is assigned one service pair per month
Model Overview

Each service has at least the minimum number of residents but no more than the Maximum, for all half months.

\[
\sum_{p \in P} X_{rpm} = 1 \quad \forall r \in R, m \in M:
\]

\[
l_{sm} \leq \sum_{p \in P_{si}} X_{rpm} \leq u_{sm} \quad \forall m \in M, s \in S, i \in 0,1
\]
Each resident has at least the minimum number of months in each service but no more than the maximum, for all months
Model Overview

\[ \sum_{p \in P} X_{rpm} = 1 \quad \forall r \in R, m \in M : \]

\[ l_{sm} \leq \sum_{p \in P_{si}} X_{rpm} \leq u_{sm} \quad \forall m \in M, s \in S, i \in 0, 1 \]

\[ l_{rs} \leq \sum_{p \in P} \sum_{m \in M} q_{ps} \times X_{rpm} \leq u_{rs} \quad \forall r \in R, s \in S \]

\[ b_t X_{rpm} + b_t X_{rp(m+1)} + b_t X_{rp(m+2)} \leq Y_{rm} + 2 \]

\( Y \) is equal to 1 if resident \( r \) works more than 3 ‘hard’ service pairs in a row
Model Overview

\[
\sum_{p \in P} X_{rpm} = 1 \quad \forall r \in R, m \in M:
\]

\[
l_{sm} \leq \sum_{p \in P_{si}} X_{rpm} \leq u_{sm} \quad \forall m \in M, s \in S, i \in 0, 1
\]

\[
l_{rs} \leq \sum_{p \in P} \sum_{m \in M} q_{ps} \times X_{rpm} \leq u_{rs} \quad \forall r \in R, s \in S
\]

\[
b_t X_{rpm} + b_t X_{rp(m+1)} + b_t X_{rp(m+2)} \leq Y_{rm} + 2
\]

\[
Z \geq \sum_{m \in M} Y_{rm} \quad \forall r \in R
\]

Z is equal to the maximum number of times a resident works 3 difficult services in a row
The amount of desired vacation requests filled in a schedule must be in between the upper and lower bounds
Model Implementation

- We coded the model into a C++ program with the help of the cplex library.
- The Chief resident submitted input files specifying the interns, services, service pairs and their upper and lower bounds.
- Our program read in the input files and then using their data produced a block schedule for the pediatric department interns.
Results

• We created a block schedule for 27 interns for this current year in under 5 minutes per run
• We saved the chief resident countless hours in schedule making
• We found a more optimal schedule than the chief resident could have on her own
Future Work

• We want to integrate all types of resident regardless of their year in the program so that we can optimize the total schedule

• We are restructuring our code to make it more user friendly and flexible to future changes
CHEPS and the HEPS Master’s Program

- **CHEPS:** The Center for Healthcare Engineering and Patient Safety
- **HEPS:** Industrial and Operations Engineering (IOE) Master’s Concentration in Healthcare Engineering and Patient Safety offered by CHEPS
- CHEPS and HEPS offer unique multidisciplinary teams from engineering, medicine, public health, nursing, and more collaborating with healthcare professionals to better provide and care for patients
- For more information, contact Amy Cohn at amycohn@umich.edu or visit the CHEPS website at: https://www.cheps.engin.umich.edu
Questions?

Peter Mayoros  
pmayoros@umich.edu

Prof. Amy Cohn  
amycohn@med.umich.edu

Department of Industrial and Operations Engineering  
University of Michigan