Assigning officers in the USS Enterprise: A Mathematical Matching Model based on Klingonian techniques

Cpt. James T. Kirk

March 27, 2017

Introduction

In this section I will explain we discuss a mathematical model for the problem by which we assign officers to duties in the spaceship USS Enterprise. Mr. Spock recommended to me that I take MAT 168 and after learning the beautiful theory of linear optimization I learned that this can be applied in this important circumstances. This was developed after I took a class at UC Davis while on leave from the star fleet.

The Problem

Every year, cadets are chosen to fill positions in the USS enterprise. Initially, this was done by first asking cadets for their preferences of duties for the coming quarter, and then having the captain match the new officers to their preferred duties. Unfortunately, this process took upwards of 8 light years to complete and the designated assignments generally failed to produce an optimal solution. Klingon's could do it in just 10 minutes using Math so we decided to finally copy the enemy on their smart ways and use Mathematical optimization tools to do so. Our main reference is the latest book by its[**?**, **?**, **?**]

Why is this interesting

Here you describe either applications or how it relates to other parts of mathematics. But the main reason we care is to keep control on the Klingons...

Me and my crew trying to convince UCD students to stop using bycicles

1 The Summary of the proposed Solution

In order to remedy this issue, we designed a model which incorporates integer programming to matching theory in order to pair all officers to duties. From now on to avoid the enemy knowing what we are up to we will call officers graduate students and fleet duties as TA ships. *(NEVER REVEAL THIS to Klingons!).*

Goal: Our ultimate goal is to assign graduate students to courses while attempting to maximize the satisfaction of the graduate students. Creating such a model required that we consider certain constraints, such as teaching assistant seniority, time conflicts, teaching ability, and happiness of each teaching assistant.

2 Technical Details

Key plenty of example equations

First we demonstrate how to enter mathematical symbols and equations within normal text. The "happiness" function be defined as: $f(\mu_i, z_i) = \mu_i - z_i + 1$ f takes a number of sections μ and a number of distinct courses z and returns how desirable that combination is. To understand it a little more clearly, observe the following table (note that μ values are in the rows and z values are in the columns):

We can start with an innocent looking integral $\int\limits_a^b x^2 dx$ inside text or later we can do multiple integrals in display mode:

$$\iint_V \mu(u,v)\,du\,dv$$

$$\iiint_V \mu(u,v,w)\,du\,dv\,dw$$

$$\iiiint_V \mu(t,u,v,w)\,dt\,du\,dv\,dw$$

$$\int \cdots \int_V \mu(u_1,\dots,u_k)\,du_1 \dots du_k$$

Summation $\sum_{n=1}^{\infty} 2^{-n} = 1$ inside text or in display mode

$$\sum_{n=1}^{\infty} 2^{-n} = 1$$

$$\prod_{i=a}^{b} f(i)$$

Can play with fractions $\int \frac{1}{2} dx$ Many other symbols can be used For example the binomial coefficient is defined by the next expression:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

And of course this command can be included in the normal text flow $\binom{n}{k}$.

- x_{ij} be the Boolean value of whether TA i is assigned to section j,

$p_{i\sigma}^{TOD}$

$$p_{i\sigma}^{TOD}\begin{cases} 100-\frac{50}{|\tau|}\psi_i^+(TOD(j)); & \text{likes } \sigma \\ 50; & \text{indifferent towards } \sigma \\ 0+\frac{50}{|\tau|}\psi_i^-(TOD(j)); & \text{dislikes } \sigma \end{cases}$$

In-line maths elements can be set with a different of format:

- $P_{ij}=\frac{e_i p_{i\sigma}^{TOD}+o_i p_{ik}^{course}}{o_i+e_i}$
- o_i depends on how important TA i finds courses.
- e_i depends on how important TA i finds TOD.

$$o_i \ and\ e_i=\begin{cases} 1\ and\ 1; & \text{indifferent;} \\ 1\ and\ 2; & \text{times more important} \\ 2\ and\ 1; & \text{courses more important;} \end{cases}$$

- $\phi_i^+(k)$ is the rank given to course k by TA i on the courses liked list.
- $\phi_i^-(k)$ is the rank given to course k by TA i on the courses disliked list.
- $\psi_i^+(\sigma)$ is the rank given to TOD σ by TA i on the times liked list.
- $\psi_i^-(\sigma)$ is the rank given to TOD σ by TA i on the times disliked list.

If σ is not on the times liked list $\psi_i^+(\sigma)=0$. If σ is not on the times disliked list $\psi_i^-(\sigma)=0$.

If k is not on the courses liked list $\phi_i^+(k)=0$. If k is not on the courses disliked list $\phi_i^-(k)=0$.

- let μ_i be the number of sections taught by TA i. So our constraint,

$$\mu_i=\sum_{j\in S}x_{ij}$$

- $R_i=c(5-r_i)$ is the rank value of TA i.
  - the fact that R_i is linearly dependent on r_i assures that better TAs (those with a better rank) are assigned the sections they prefer
  - c is a constant, currently c=10
- z_i is the number of distinct courses TA i will teach.It is calculated as follows:

$$z_i=\sum_{k\in C}q_{ik}$$

where $q_{ik}\in\mathbb{Z}$, $0\le q_{ik}\le 1$, and

$$q_{i,k}\begin{cases} \ge \sum_{j\in S}\frac{x_{ij}}{maxunits}\delta^{course}(j,k); & a_i\ge 0 \\ \le \sum_{j\in S}x_{ij}\delta^{course}(j,k); & a_i<0 \end{cases}$$

where

$$\delta^{course}(j,k) = \begin{cases} 1; & course(j) = k \\ 0; & course(j) \neq k \end{cases}$$

$$v_i = \sum_{d \in D} y_{id}$$

where $y_{id} \in \mathbb{Z}$, $0 \leq y_{id} \leq 1$, and

$$y_{id} \begin{cases} \geq \sum_{j \in S} \frac{x_{ij}}{maxunits} \delta^{day}(j,d); & b_i \geq 0 \\ \leq \sum_{j \in S} x_{ij} \delta^{day}(j,d); & b_i < 0 \end{cases}$$

where

$$\delta^{day}(j,d) = \begin{cases} 1; & day(j) = d \\ 0; & day(j) \neq d \end{cases}$$

We want to choose $y_{id} = 0$ if possible since as y_{id} increases, so does w_i, and w is negative in $h(\mu, w)$. This is true when b_i is positive.

- w_i is the number of pairs of back to back sections that are assigned to TA i.It is calculated as follows:

$$w_i = \sum_{(j,j') \in S \times S} \beta_{ijj'}$$

  - where $\beta_{ijj'} \in \mathbb{Z}$ and is subject to

$$\frac{1}{2}\delta^{btb}(j,j')(x_{ij} + x_{ij'}) \geq \beta_{ijj'} \geq \frac{1}{2}\delta^{btb}(j,j')(x_{ij} + x_{ij'} - 1)$$

  - where

$$\delta^{btb}(j,j') = \begin{cases} 1; & 10min \leq t_{start}(j) - t_{end}(j') \leq 30min \\ 0; & otherwise \end{cases}$$

$\delta^{btb}(j,j')$	x_{ij}	$x_{ij'}$	Lower bound on $\beta_{ijj'}$	Upper bound on β	$\beta_{ijj'}$
0	0 or 1	0 or 1	0	0	0
1	1	0	0	$\frac{1}{2}$	0
1	0	1	0	$\frac{1}{2}$	0
1	0	0	0	$-\frac{1}{2}$	0
1	1	1	$\frac{1}{2}$	1	1

$$\begin{bmatrix} \mu \backslash z & 1 & 2 & 3 & 4 \\ 1 & 1 & - & - & - \\ 2 & 2 & 1 & - & - \\ 3 & 3 & 2 & 1 & - \\ 4 & 4 & 3 & 2 & 1 \end{bmatrix}$$

$$\begin{pmatrix} \mu\backslash w & 0 & 1 & 2 & 3 \\ 1 & 0 & - & - & - \\ 2 & -1 & 1 & - & - \\ 3 & -2 & 0 & 2 & - \\ 4 & -3 & -1 & 1 & 3 \end{pmatrix}$$

We want to maximize

$$\sum_{i \in T} R_i(a_i * \gamma_1 f(\mu_i, z_i) + b_i * \gamma_2 g(u_i, w_i) + c_i * \gamma_3 h(\mu_i, v_i) + \sum_{j \in S} x_{ij} p_{ij})$$

Where γ_1, γ_2, and γ_3 are constants that will be determined experimentally. The following are the constraints placed on the objective function $\forall j \in S$, $\sum_{i \in T} x_{ij} = 1$ also $\forall i \in T$ $\sum_{j,j' \in S \times S} \Phi_{ijj'} = 0$ where $\Phi_{ijj'} \in \mathbb{Z}$ such that

$$\frac{1}{2}\Omega^{overlap}(j,j')(x_{ij} + x_{ij'}) \geq \Phi_{ijj'} \geq \frac{1}{2}\Omega^{overlap}(j,j')(x_{ij} + x_{ij'} - 1)$$

- subject to lower division TAs:

  $\forall (i,j) \in LoTA \times (Upsec \cup Gradsec)$, $x_{ij} = 0$

  - No TA unsuited for upper division classes teaches upper division classes

- subject to non graduate TAs:

  $\forall (i,j) \in (LoTA \cup UpTA) \times Gradsec, x_{ij} = 0$

  - No TA unsuited for graduate classes teaches graduate classes

- Phantoms

  - The Phantom TA and the Phantom Section are not subject to any of the constraints above.

3 In Depth investigation: What is the mathematical meaning of "Happiness"

Our assignment will try to maximize the happiness of students, so we have to discuss in detail how we defined that and argue that this makes sense. We also attempt to disperse the happiness fairly among the TAs The objective function will be subject to some conditions $\forall i, k \in T \times T$, and

$$|\sum_{j \in S} \frac{x_{i,j} p_{i,j}}{u_i} - \sum_{j \in S} \frac{x_{k,j} p_{k,j}}{u_k} | \leq \Delta_{happy}(|r_i - r_k| + 1),$$

where we define Δ_{happy} to be as small as the IP allows without making the system infeasible

SCIP

- The Number of Parameters
  - For a section j the following parameters must be determined:
    * $\forall k \in C$: $\delta^{course}(j,k)$
    * $\forall d \in D$: $\delta^{day}(j,d)$
    * $\forall j' \in S$: $\delta^{btb}(j,j')$, $\Omega^{overlap}(j,j')$
  - For a TA we must also obtain all p_{ij}'s and R_i's
  - The number parameters going into SCIP is given by:

$$|S|(|C| + |D| + |S|) + |T|(|S| + 1)$$

- The Number of Variables in a Solution
  - In a given solution SCIP must calculate the following:
    * x_{ij} $\forall(i,j) \in T \times S$
    * q_{ik} $\forall(i,k) \in T \times C$
    * y_{id} $\forall(i,d) \in T \times D$
    * $\beta_{ijj'}$ $\forall(i,jj') \in T \times (S \times S)$
  - So a solution has to assign values to $|T|(|S| + |C| + |D| + |S|^2)$ variables.
- The Number of Constraints
  - $|T|$ constraints to ensure each TA meets his or her teaching duties.
  - $|S|$ constraints to ensure each section has one TA.
  - $|T| * |S|^2$ constraints ensure that no TA is assigned classes that occur at the same time.
  - $|LoTA|(|UpSec| + |GradSec|)$ constraints ensure no TA unfit to teach Upper division and graduate level courses does so.
  - $|UpTA||GradSec|$ constraints ensure no TA unfit to teach graduate courses does so.
  - We also have to take into account constraints due to TA schedule conflicts. Let this number of conflicts be Q.
  - The total number of constraints is about $|T| + |S| + |T| * |S|^2 + |LoTA|(|UpSec| + |GradSec|) + |UpTA||GradSec| + Q)$

In an optimal solution, if TA i does not feel indifferent towards course k, q_{ik} always returns 1 when at least one of the sections ,j, i is assigned to has $course(j) = k$ and 0 otherwise.

proof (by contradiction): Assume that M is an optimal solution to our objective function and that q_{ik} does not always return 1 when at least one of the sections ,j, i is assigned to has $course(j) = k$ and 0 otherwise.

- case 1: i prefers having as many different courses as possible. $a_i > 0$
  - case 1: $\exists j_0$ such that $x_{ij_0} = 1 \wedge \delta^{course}(j_0, k) = 1$ but $q_{ik} = 0$
    - this is a contradiction since $q_{ik} \geq \sum_{j \in S} \frac{x_{ij}}{4} \delta^{course}(j, k)$
  - case 2: $\neg\exists j_0$ such that $x_{ij_0} = 1 \wedge \delta^{course}(j_0, k) = 1$ but $q_{ik} = 1$
    - this is a contradiction since $q_{ik} = 0$ results in a solution M' that is larger than M.
- case 2: i prefers having as few courses as possible. $a_i < 0$
  - case 1: $\exists j_0$ such that $x_{ij_0} = 1 \wedge \delta^{course}(j_0, k) = 1$ but $q_{ik} = 0$
    - this is a contradiction since $q_{ik} = 1$ results in a solution M' that is larger than M.
  - case 2: $\neg\exists j_0$ such that $x_{ij_0} = 1 \wedge \delta^{course}(j_0, k) = 1$ but $q_{ik} = 1$
    - this is a contradiction since $q_{ik} \leq \sum_{j \in S} x_{ij} \delta^{course}(j, k)$

We can create a new page easily too!

4 Appendix: Code and Data

After having created a model, we attempted to implement it. This required that we design a website that had a user interface, create a database to store the information gathered, and write code that implemented our model.

References

[1] Leonard McCoy. Space travel and why it is such a hassle. *Journal of the Star Academy*, 69:19–150, 3962.

[2] William Shattner. *The Star Trek universe*, volume 11 of *Cool books that do not exist series*. Cambridge Univ. Press, San Francisco, 3989.

[3] ?? Spock. The speed of light, einstein was my co-pilot. *Quantum Physics Monthly*, 69:9–15, 1962.