A Decision Support Tool to Automate and Optimize Contract Staffing using Linear Programming

Paris Joslin (Team Leader), Zachary Leondike, Conner Oxford, Nathan Skinner, Will Cunningham
Stephen Sieck (Senior Program Manager), Nick Marquart (Co-Founder and Chief Analytics Officer)

Infinity Labs LLC

Infinity Labs is an innovation firm specializing in defense contracting in the areas of simulation and cyber security. Infinity Labs has four main groups of employees ranging from executives, program managers, project managers, and LCAT (Labor Category) employees. Our system of interest is the staffing process at Infinity Labs.

Current Staffing Process

The current staffing process is manual and time consuming. Program managers begin by looking at all current employee utilizations. Staffing meetings are then held to determine an employee's eligibility, and assignments are created based on the situational knowledge of the program manager.

Staffing Decision Support Tool

Our team created an Excel-based decision support tool that works in parallel with the optimization model.
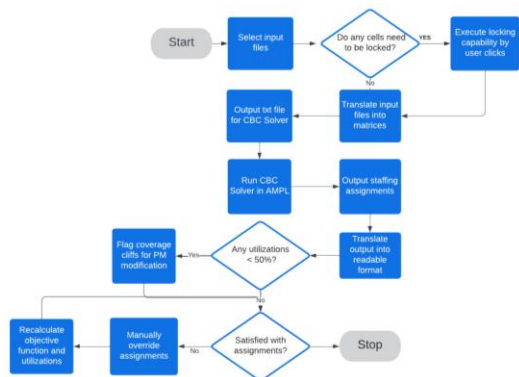
The interface provides a simple method for uploading input data, locking current assignments, running the model, and making any post-hoc decisions.

Balancing Utilization using Optimization

We created an optimization model using VBA and the CBC solver in AMPL to create employee assignments. This model minimizes employee utilization variability while ensuring all constraints are met:

Objective:
$$Minimize\ Z - W + \sum_{j=1}^{J} 5*F_j + \sum_{j=1}^{J} 5*K_j + \sum_{s=1}^{S} 100*D_s + \sum_{j=1}^{J} 100*H_j$$

Parameters:

S = Employee
J = Contract
E_j = Budget per contract
N_s = LCAT rate per employee
P_s = Number of projects per employee
$G_{s,j}$ = Clearance eligibility
$L_{s,j}$ = Employee location eligibility
R_j = Recommended hours per contract
C_s = Maximum hours per employee

Decision Variables:

$X_{s,j}$ = Binary - If an employee is assigned to a contract
$Y_{s,j}$ = Hours assigned per employee per contract
U_s = Dummy variable for employee utilization
Z = Maximum utilization
W = Minimum utilization
F_j = Penalty for understaffing projects
K_j = Penalty for surpassing budget
D_s = Penalty for violating employee assignments
H_j = Penalty for violating contract assignments

$$\sum_{s=1}^{S} Y_{s,j} \geq R_j - F_j$$

For every employee s and contract j, the sum of hours per contract must exceed the recommended hours with a penalty if a contract needs to be understaffed.

$$\sum_{j=1}^{J} X_{s,j} \leq P_j$$
$$\sum_{j=1}^{J} Y_{s,j} \leq C_s$$

For every employee s, the number of allowable projects and maximum workable hours must not be exceeded.

$$L_{s,j} \geq X_{s,j}$$
$$G_{s,j} \geq X_{s,j}$$

For every employee s and contract j, the employee must have the correct location and clearance level in order to be assigned to a contract.

$$\sum_{s=1}^{S} N_s Y_{s,j} \leq E_j + K_j$$

For every employee s and contract j, the cost of assignments must not exceed the contract budget with a penalty if the budget is surpassed.

Other constraints not shown above include eligibility, locked assignments, max hours, max projects, and utilization calculation constraints.

Assignments and Contract Analysis

Our tool merges the model output with any decisions made in the VBA tool to portray a three-month outlook of new assignments.


Our tool also analyzes how post-processing decisions affect contract budgets and employee utilizations. Program managers can use the utilization analysis to address coverage cliffs.

Results and Impact

When comparing employee utilizations both before and after tool implementation, our team concluded that there was a 17.3% increase in average utilizations.. All employees are now balanced at 99% utilization when overriding previous assignments.
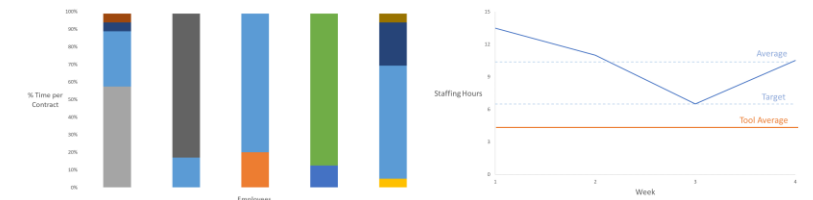
Our tool reduced the utilization variability by 115%. Our tool also saved 5.9 hours on average per week for program managers, while increasing stakeholder satisfaction by 43%.