DESTINATION: DEEP SPACE

Score Optimization with Linear Programming

by

Ed Law, Ph.D.
Retired Senior Engineering Specialist
Fiat Chrysler Automobiles

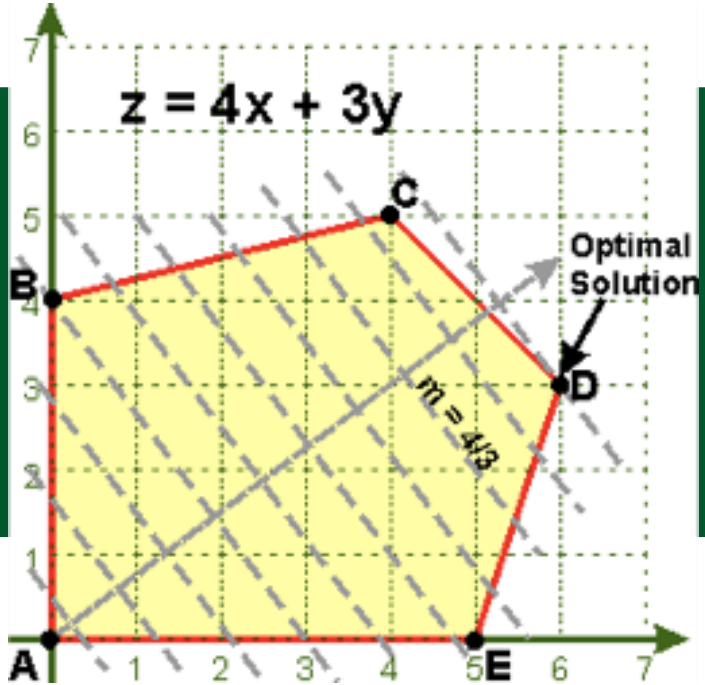
Mentor, FRC Team 8 (Paly Robotics)
Former Lead Mentor and Coach, FRC Team 2834 (Bionic Black Hawks)

What is the purpose of score optimization?

Using scouting data collected, the program can find the best combination on which alliance partner should score hatches and cargos at what locations in order to maximize score. This enables us to:

- Assign robot tasks – knowing quickly who should do what in a match gives a good starting point for fine tuning the match tactics.
- Adjust match tactics – predicting the outcome of a match with no defense allows us to adjust tactics if needed.
- Establish defensive tactics – knowing what the opposing alliance may probably do allows us to establish defensive plans if needed.
- Help to create a pick list for alliance selection – trying different scenarios and predicting match outcome of each helps to pick the right partners.

What is linear programming?



Linear programming is a mathematical technique for the optimization of a linear objective function, subject to linear equality and inequality constraints. An example of a two variable system is shown on the left.

How can linear programming be used in the 2019 FRC Game?

- In the DESTINATION: DEEP SPACE 2019 FRC game, the objective function is the total score of which we want to maximize.
- The constraints are the limits of the total number of scoring positions of hatch panels and cargos and the capabilities of each alliance partner. An additional constraint is that the total number of cargos scored must be less than or equal to the total number of hatch panels scored plus null hatches.
- However, the desire to complete a rocket to earn extra Ranking Points makes this into a nonlinear programming problem because of the dependencies of the different variables.
- The drawback of a nonlinear programming problem is that the solution cannot be guaranteed to be in the true optimal condition.

How can linear programming be used in the 2019 FRC Game?

- To remove the nonlinearity due to the rocket, we change the problem into a two-step linear programming problem.
- First we check to see if a rocket can be achieved by the alliance. If yes, then we will use that solution. If not, then we try to find a solution without attempting to score the level 2 and level 3 of the rocket.

How was this implemented?

- Microsoft Excel Solver was used to solve the linear programming problem using the Simplex Method. To use the Solver in Microsoft Excel, you need to activate the Add-Ins called Solver-Addin.
- A match schedule would be loaded and stored in one of the sheets. This allows us to change a match number and automatically populate the data of each of the 6 robots in that match.
- Scouting data would be loaded from a remote server using a Python script.
- A VBA script was written to automate the tasks of doing all the calculations.

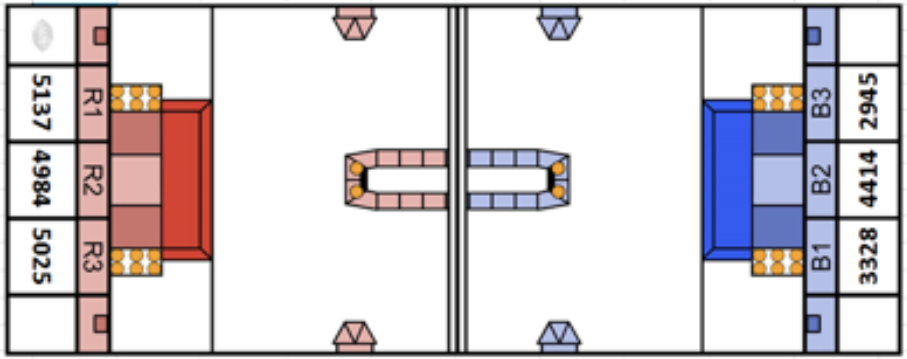
How was this implemented?

- The potential sandstorm bonus and climb bonus of each robot were included to get the predicted final match score if there is no defense.
- This was tested at our first FRC event in Week 1 at the Del Mar Regional. Since then we made some improvements to calculate how many null hatches we should put in to maximize score.

Screenshot of the Spreadsheet

1	MATCH	56		Solve		Update		
2								
3	Scouting Data		R1	R2	R3	B1	B2	B3
4		Teams	5137	4984	5025	3328	4414	2945
5	Sand Storm Bonus		6	6	3	3	6	3
6	Hatch	L1 max	2.0	3.0	4.0	2.0	3.0	1.0
7		L2 adj avg	0.0	0.0	0.3	0.0	1.0	0.0
8		L3 adj avg	0.0	0.0	0.0	0.0	1.3	0.0
9	Cargo	L1 max	3.0	3.0	6.0	3.0	4.0	5.0
10		L2 adj avg	0.0	0.0	0.0	0.0	0.7	0.0
11		L3 adj avg	0.0	0.0	0.0	0.0	0.8	0.0
12	Climb Bonus		6	3	3	3	12	3
13	Max	Cycles	5	4	7	3	11	4
14	Climb Bonus Adjust			0			0	
15	Target		R1	R2	R3	B1	B2	B3
16		Teams	5137	4984	5025	3328	4414	2945
17	Hatch	Level 1	0	0	5	0	0	2
18		Level 2	0	0	0	0	2	0
19		Level 3	0	0	0	0	2	0
20	Cargo	Level 1	5	4	2	3	3	2
21		Level 2	0	0	0	0	2	0
22		Level 3	0	0	0	0	2	0
23								
24	Score =		70			78		

Main | Red | Blue | MatchSchedule | ScoutingData



Field picture courtesy of FRC Team 1678

Null hatches use =	6	6
Rocket =	no	yes

Acknowledgement

- Caleb T. (strategy and scouting lead) for the excellent scouting app he developed to capture the scouting data and the python script to query the data from the server and export in the format for the Excel spreadsheet.
- Emma J. (strategy and scouting lead) for her active participation in the discussion and strategic insight.
- Bryan Y. and Robbie S. (drive team) for their helpful discussions and input in the development and using the spreadsheet to help with pre-match strategy.
- Annie C. (business lead) for reformating the whole presentation to fit the Team 8 brand and image.