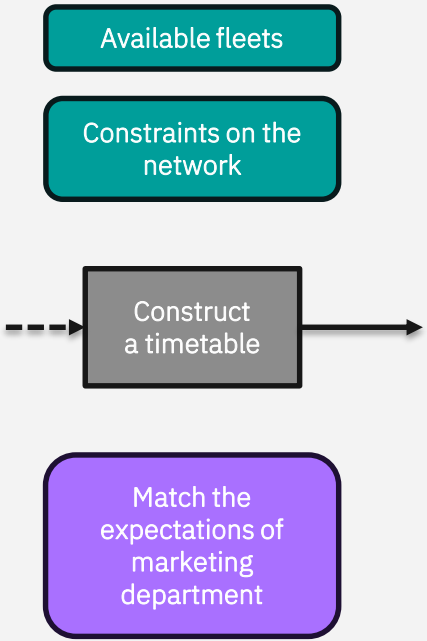
How to Approach Large-Scale Aircrew Scheduling Problems

Alexander Pozdneev
IBM

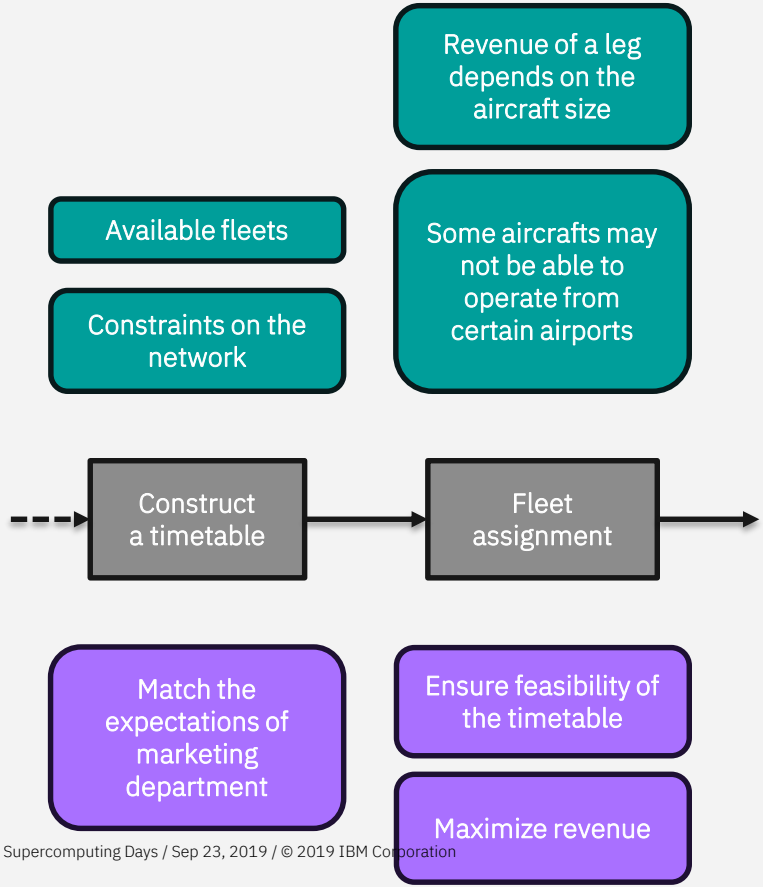
«Next to fuel costs, **crew costs** are the largest direct operating costs of airlines»

Commonly accepted industry truth

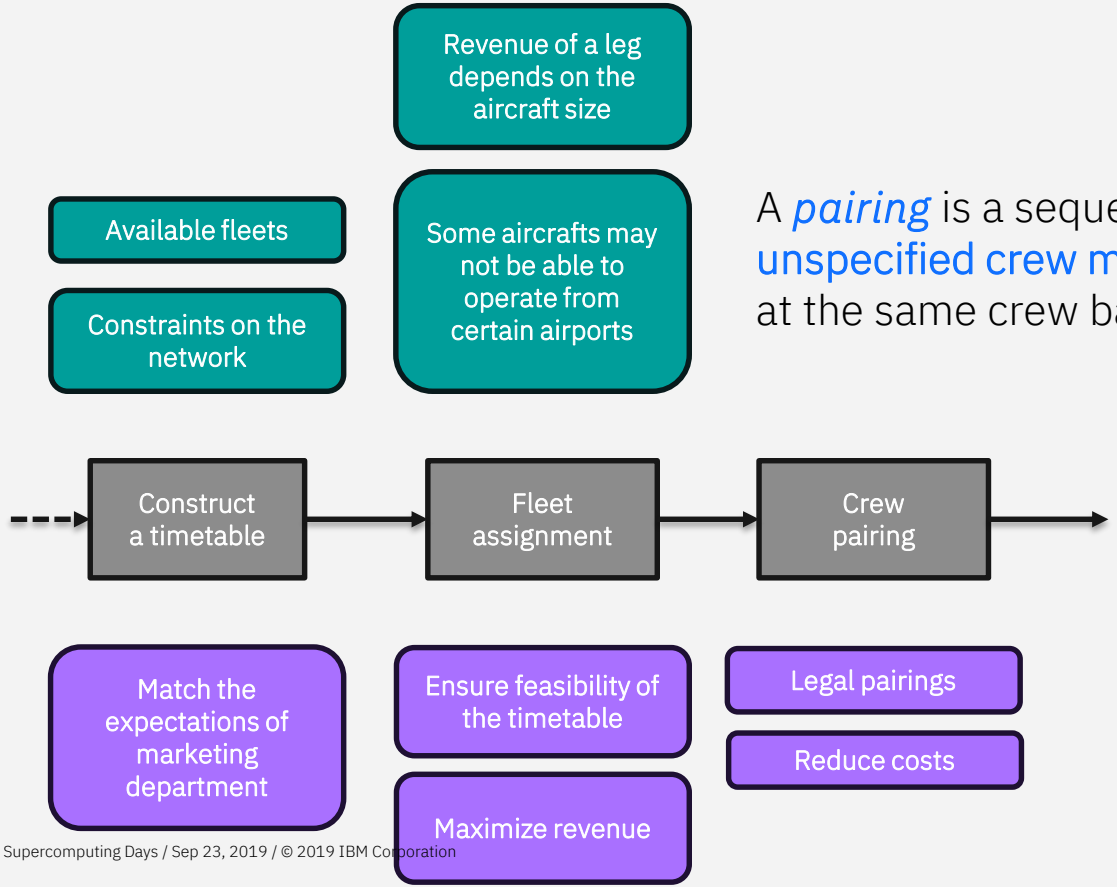
The resource planning process: Timetable



The resource planning process: Fleet

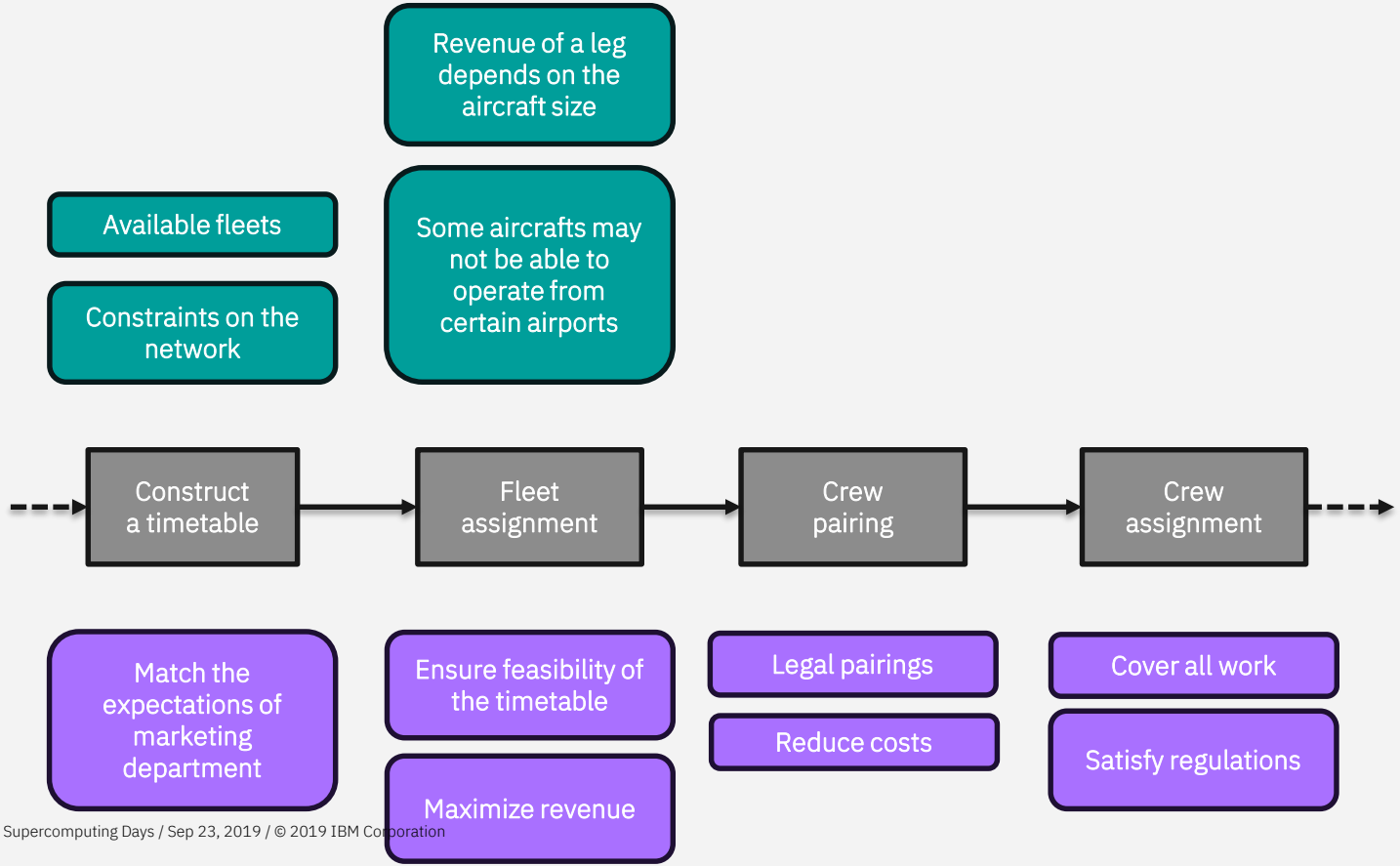


The resource planning process: Covering flight legs

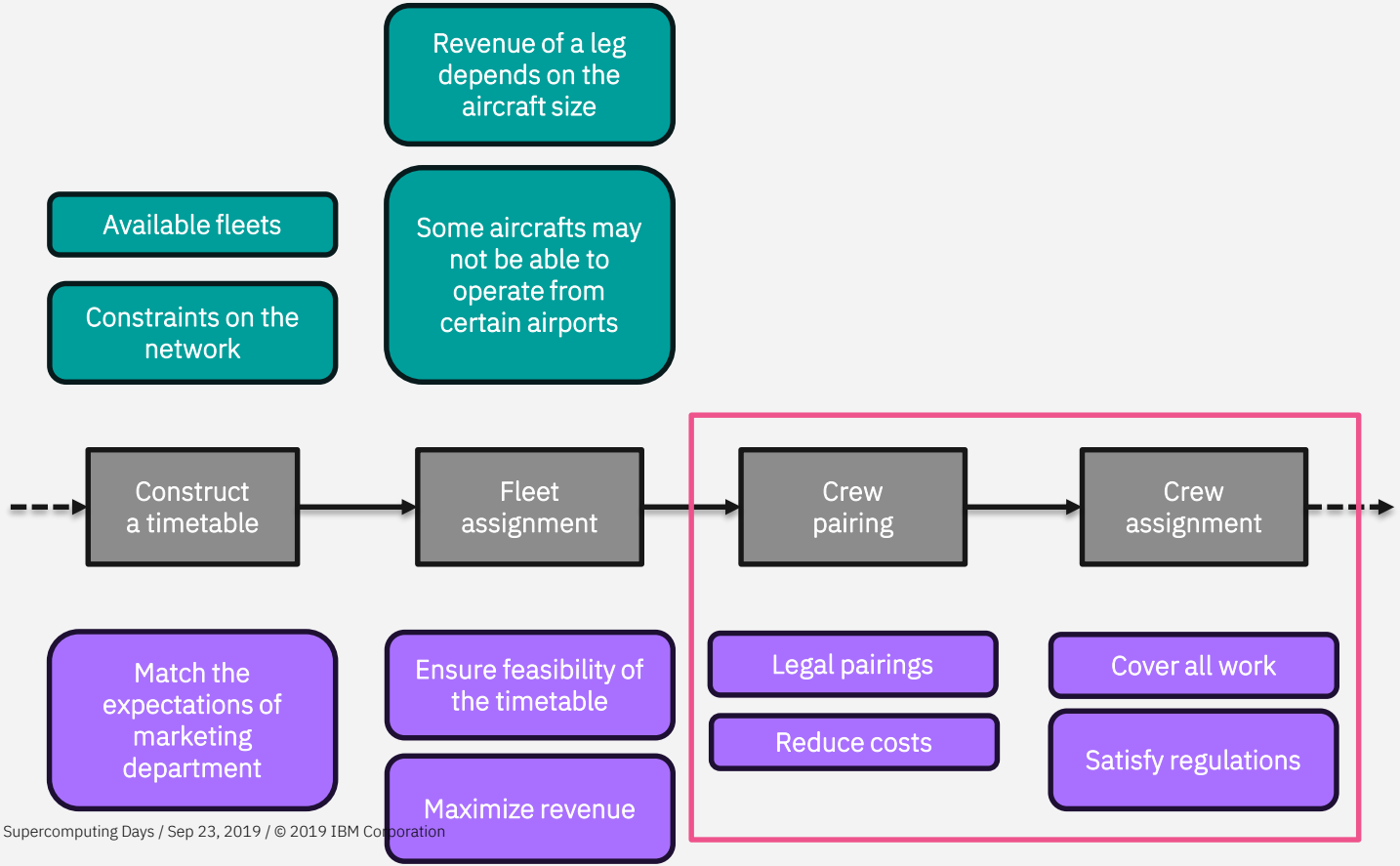


A *pairing* is a sequence of flight legs for an unspecified crew member starting and ending at the same crew base

The resource planning process: Roster



The resource planning process: Our solution



Why is that difficult?

Why is that difficult?

Covering all work

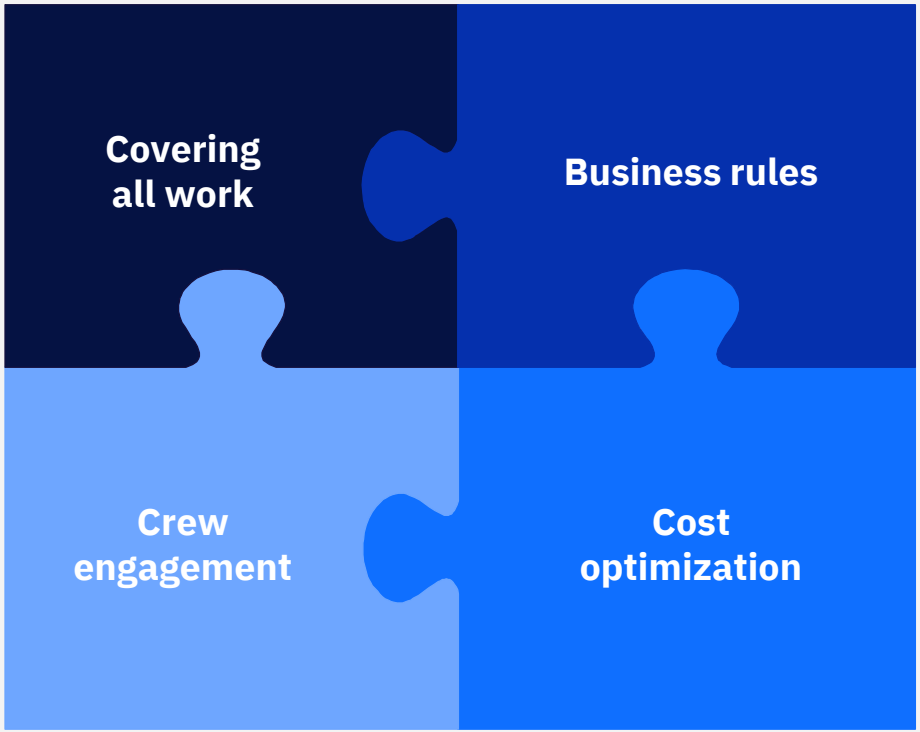
- All flight legs
- All pairings

Business rules

- Industry regulations
- Labor unions

Crew engagement

- Fairness in the work distribution



What do business users care about?

Personal requests

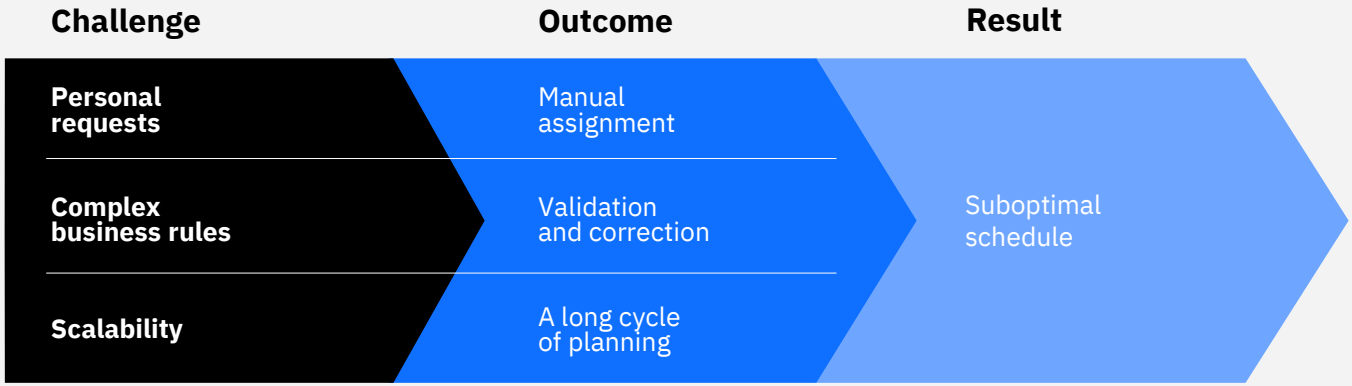
- Seniority
- Fairness

Complex business rules

- Not included into optimization
- Violation report

Scalability

- Solution by steps



How to enter new business rules?

Optimization framework by IBM Research Haifa

- Corporate platform for optimization problems
- Designed to be used by a subject matter expert
- Flexible business rules model
- Modular design
- Scalable optimization
- Web-interface
- Visual analytics tools

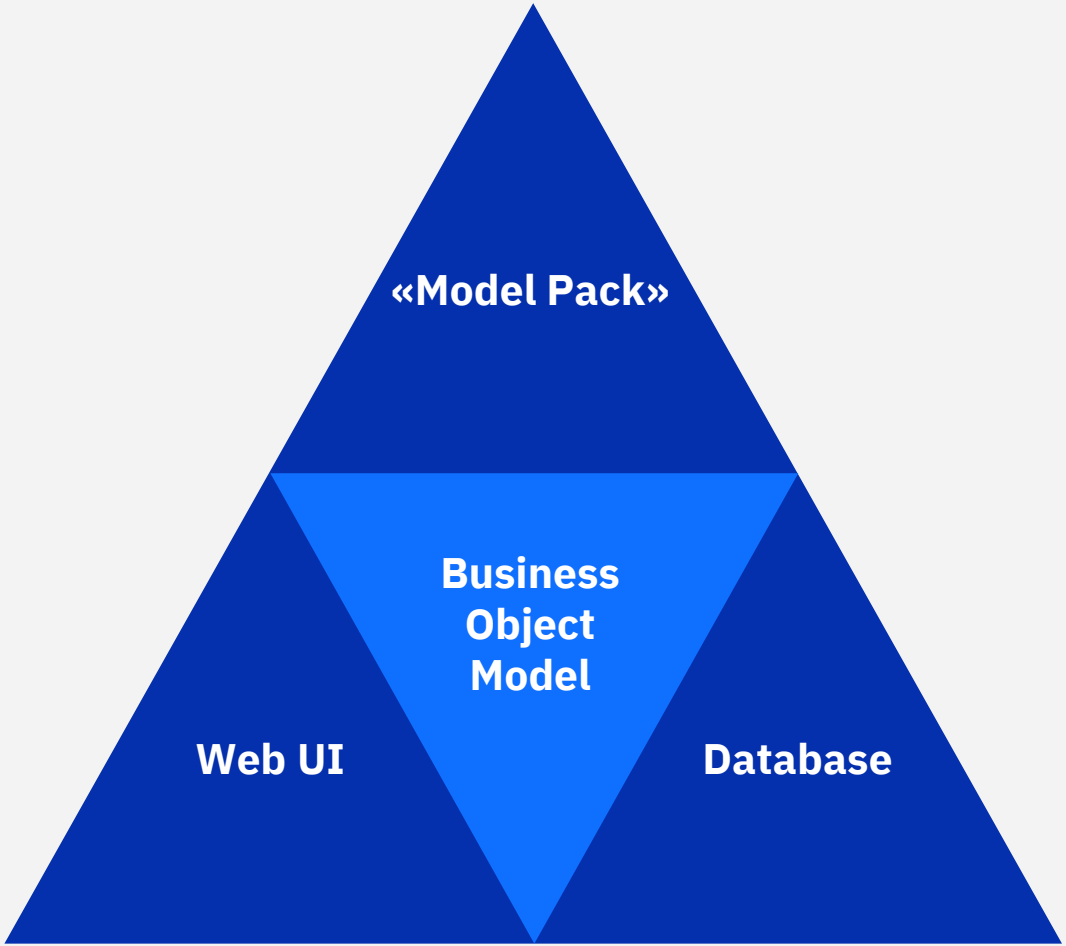
Architecture

Business Object Model
- Resources (dictionary)
- Business rules
- Periodic data

Database
- JPA

Web UI
- Domain specific visualization
- Auto generation

«Model Pack»
- Linear program abstraction
- Plugins
  - Pairing Modeler
  - Rostering Modeler
  - Railway car flow Modeler
  - VRP Modeler
  - ...
- Interface to a MILP solver

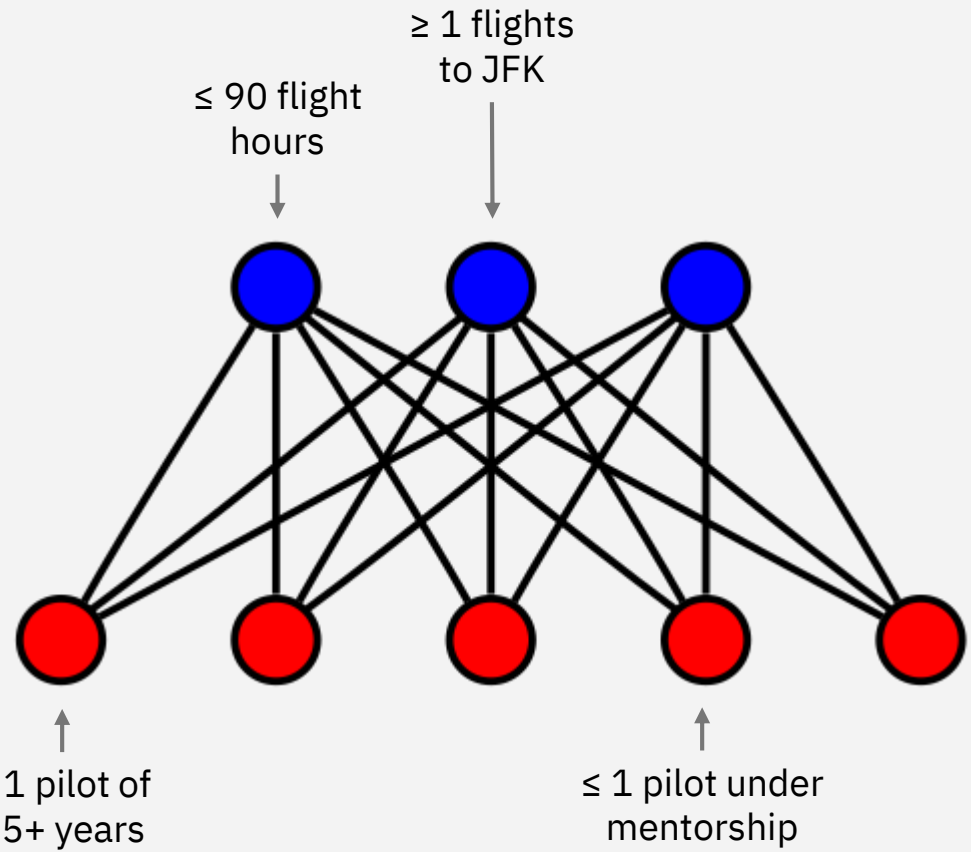


Aircrew rostering as a graph edge subsets problem

Crew members →

Assignments [*of crew members to pairings*] →

Pairings →

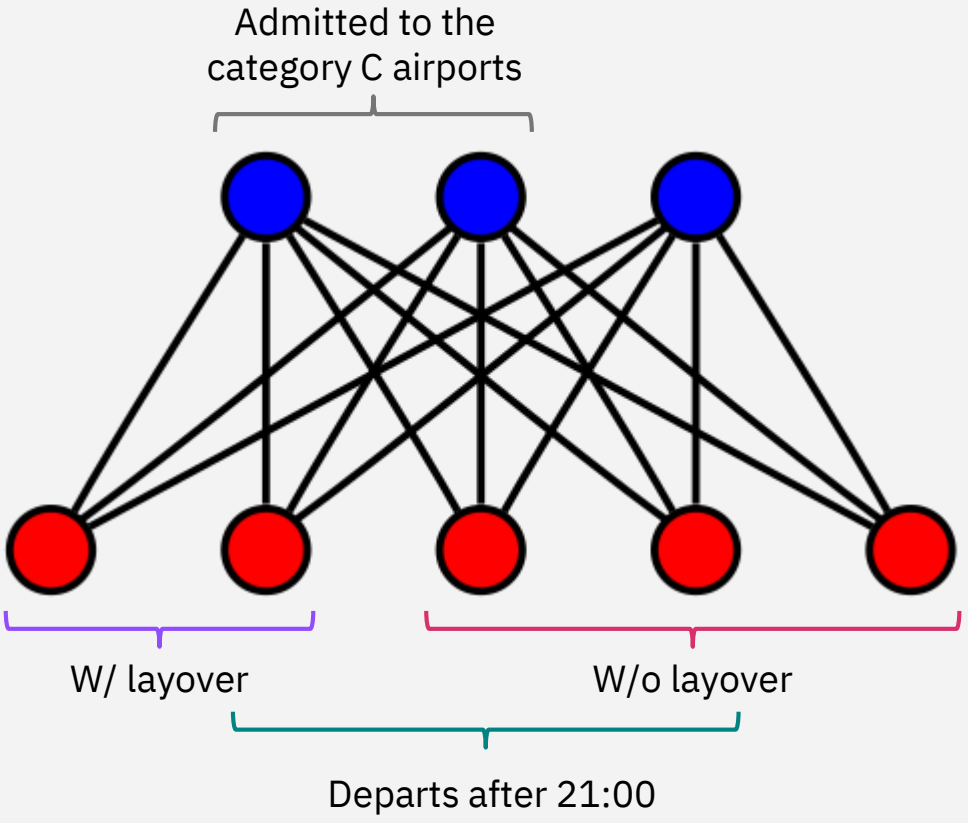


Classes of aircrew members and pairings

Crew members →

Assignments [*of crew members to pairings*] →

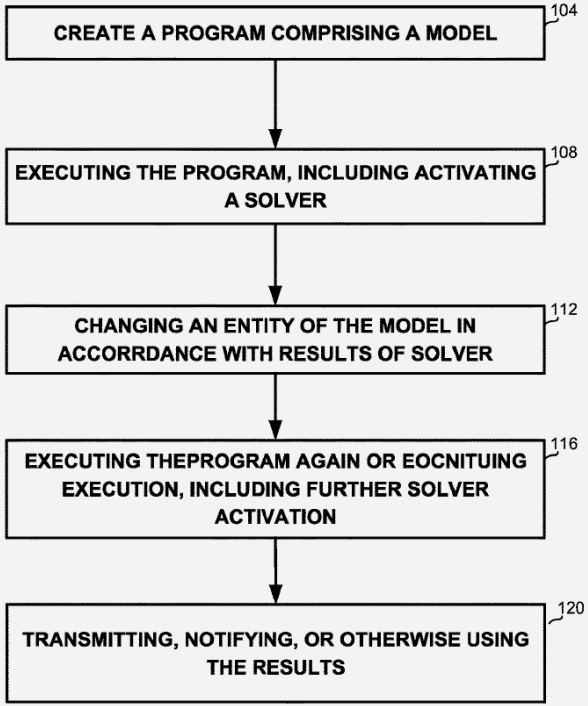
Pairings →



Reusable modeling for solving problems

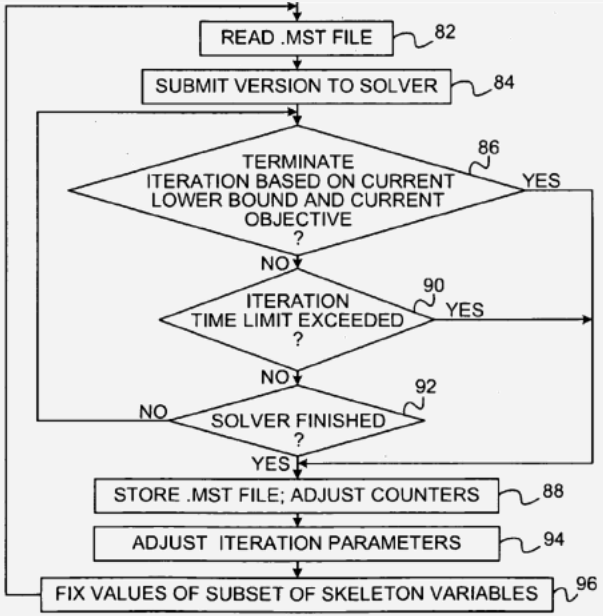
Patent US10169291B2 (2019)

- Build a model (MILP program) based on input data and business rules
  - Categorize [*decision*] variables, penalties, and equations
- Solve a model
  - Iteratively adjust and solve



Cruncher:
An MIP Solver Accelerator

Patent US20100299291A1 (2010)



Problem	Problem Size	Cplex Results	Cruncher Results
Flow Shop	10 Jobs, 10 tasks per job, 10 machines 1002 Vars, 1451 Rows 3803 non-zero elements	Objective: 951 Time: 17406 sec.	Objective: 943 Time: 137 sec.
Traveling Salesman	97 Points 9218 Vars, 9313 Rows 54722 non-zero elements	Objective: 577 Time: 1239 sec.	Objective: 519 Time: 499 sec.
Class Assignments	310 students, 7 classes 8691 Vars, 5245 Rows 46558 non-zero elements	Obj: 240000, 2000 Sec. Obj: 20000, 6200 Sec.	Objective: 20000 Time: 376 sec.
Vessel Load Planning	1825 containers, 10 destinations, 3 ports look ahead 13050 Vars, 10166 Rows 82506 non-zero elements	1000 sec. Obj. = 131936 2000 sec. Obj. = 7445 3000 sec. Obj. = 181 4000 sec. Obj. = 172	1000 sec. Obj. = 3387 2000 sec. Obj. = 845 3000 sec. Obj. = 634 4000 sec. Obj. = 578
Scheduling	14 items 7500 slots 92787 Vars, 168950 Rows 594897 non-zero elements	Objective: 90 Time: 2000 sec.	Objective: 70 Time: 220 sec.

Q&A and Contacts

Alexander Pozdneev
IBM

https://twitter.com/Alex_Pozdneev

Q&A

Conclusion

- Aircrew planning
  - Pairing optimization
  - Assignments
- Optimization framewok
  - Categorize entities
  - Create and manage domain-specific optimization flow
- Benefits
  1. Flexible business rules model
  2. End user is a domain area expert
  3. High scalability