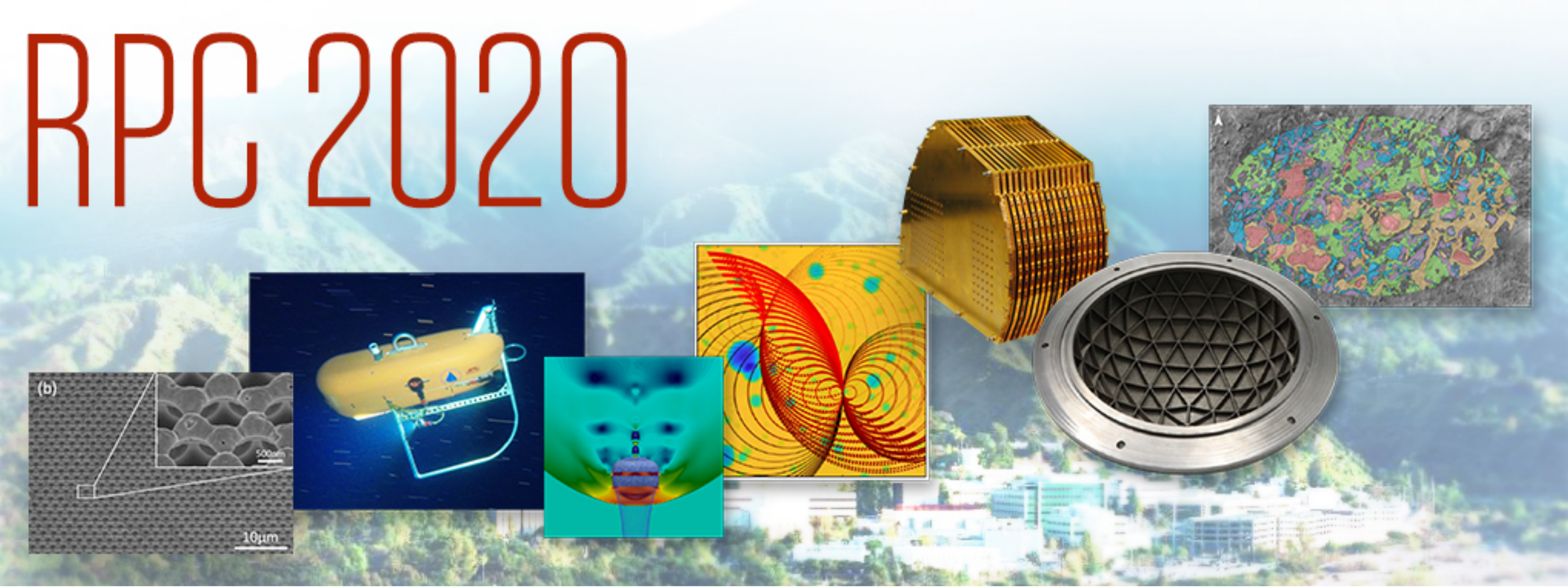


Virtual Research Presentation Conference

Automating DSN Scheduling Using Quantum Computing & Deep Reinforcement Learning

Principal Investigator: Brian Wilson (174B)
Co-Is: A. Guillaume, E. Goh, H. Venkataram, S. Alimo, M. Johnston, S. Sahnoune
Program: Strategic Initiative, 9X Quantum Scheduling

RPC-124

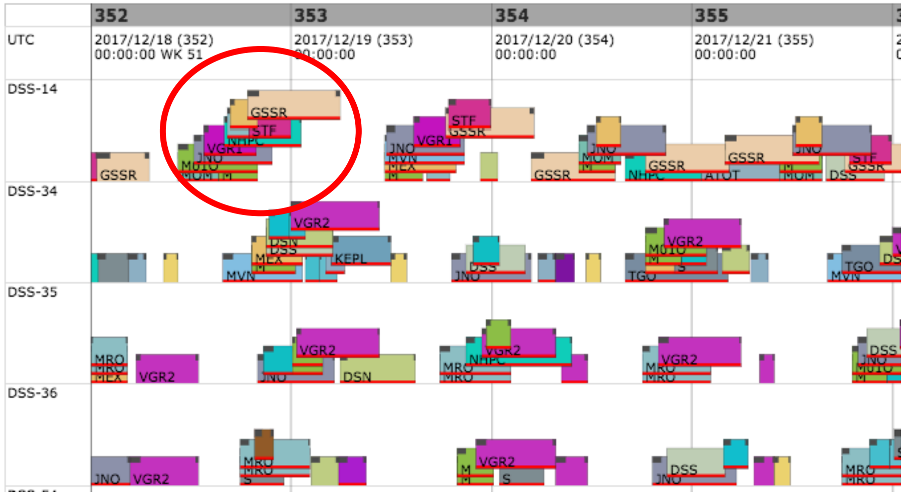
Jet Propulsion Laboratory
California Institute of Technology

Team and Collaborators

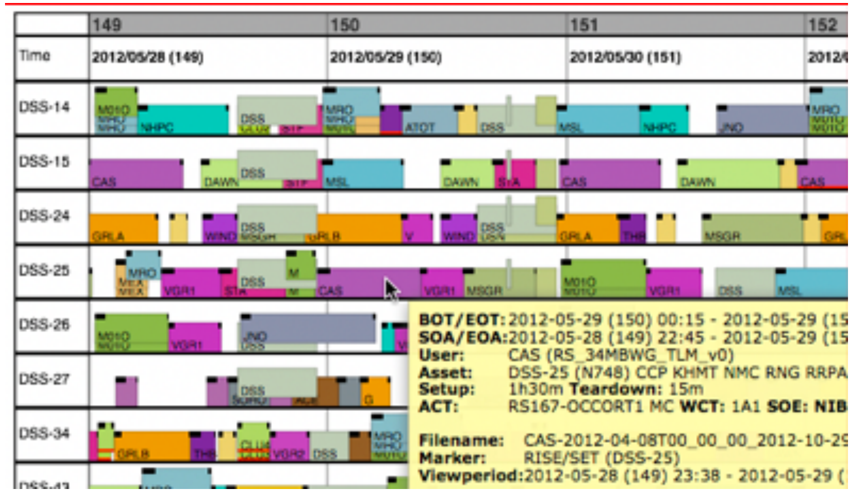
- Team
  - PI Brian Wilson (174B)
  - Co-I Alexandre Guillaume (398K): Quantum Computing Lead
  - Co-I Edwin Goh (174B): Deep Reinforcement Learning
  - Co-I Hamsa Venkataram (174B): Deep Reinforcement Learning
  - Co-I Sharouz “Ryan” Alimo (393K) & post-doc: Nonconvex optimization, DL
  - Co-I Mark Johnston (9270): Scheduling expert, impacts/assessment
  - Co-I Sami Sahnoune (174B): Schedule visualization
- Collaborations with:
  - Daniel Lidar and others at USC (Adiabatic QC Group), for Fujitsu Digital Annealer
  - Fred Glover at Meta-Analytics, for Alpha-QUBO solver
  - Microsoft QIO group, developing custom solver for us
  - Jonathon Sabol and Dr. Ramtin Madani at UT Arlington, for advanced MILP solver

The Challenge

- DSN tracks many spacecraft
- At times, antennas are very over-subscribed, up to 60%
- Automated scheduling leaves 'conflicts' that humans resolve
- Formulate as a Multi-objective, Combinatorial Optimization problem
- To resolve conflicts:
  - Move, split or shorten tracks
  - Use priorities
  - Humans "relax requirements" and missions negotiate tradeoffs
- Mission health & safety and science return are paramount



S³ tool showing "conflicting" tracking requests before any scheduling resolution



Remaining conflicts after multi-pass scheduling

Combinatorial Optimization Problem

- Algorithm Formulation
  - Binary integer variables (0 or 1)
  - Discrete time slots

$$x_{ijk} = 1$$

means
Spacecraft *i* scheduled on
Antenna *j* for
Time slot *k*

SC #1 SC #2

0	0	1	1	1	1	1	0	0	1	1	1

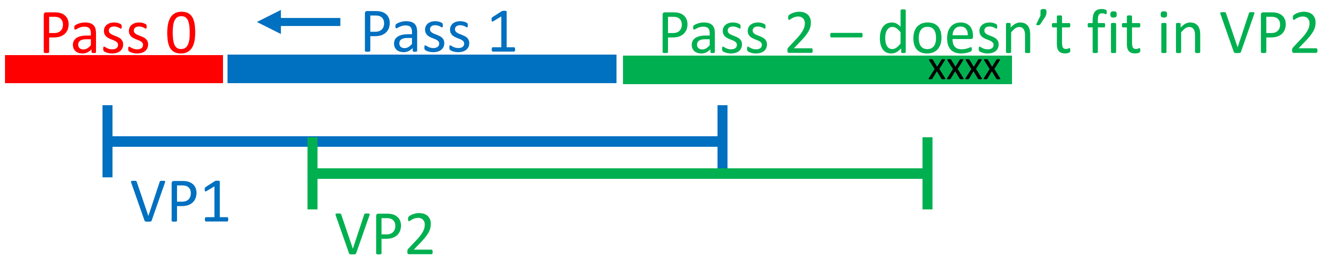
5 or 15-minute time slots

- Large Set of Request Types and Constraints
  - One spacecraft tracked in each time slot (except for Mars)
  - Track durations less than request are penalized
  - Required resources: 34m or 70m antenna, uplink/downlink
  - Coherent Arraying: Use 2, 3, or 4 34m antennas together

The Slotting Game

- Place spacecraft tracking passes to satisfy mission requests (User Loading Profiles or ULP's)
  - Subject to view periods, and requested resources

- When conflicts arise, apply strategies to resolve them:
  - Move a pass to another antenna or another complex
  - Slide a pass to accommodate another spacecraft
  - Shorten a pass to accommodate another spacecraft
  - Split a pass, but retaining segments of sufficient length
  - Drop a pass, to give priority to a critical event

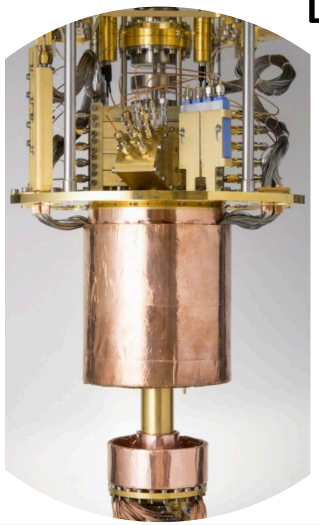


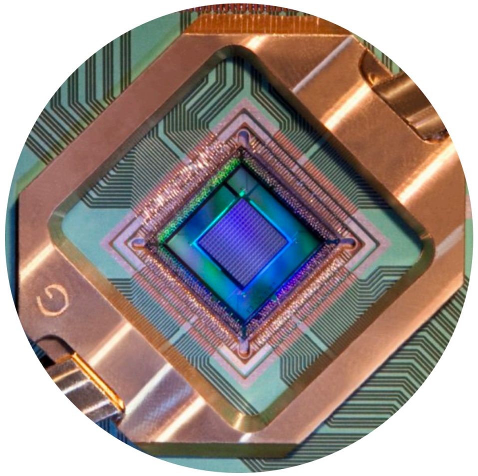
Overlapping View Periods

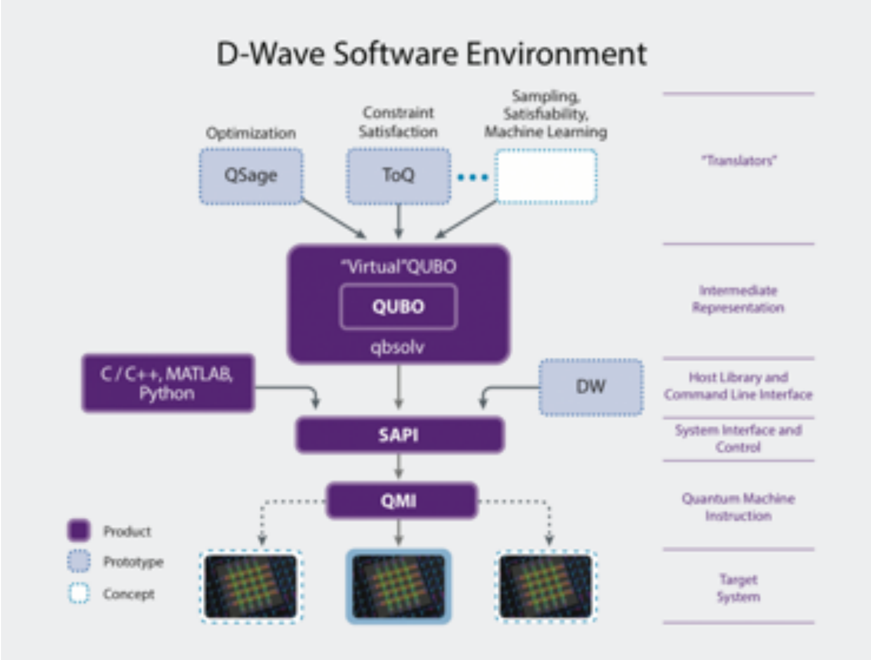
Quantum Computing

- Two main approaches:
  - Circuits of Quantum Gates (IBM, Microsoft, Google, Rigetti)
  - D-Wave Quantum Annealing
- D-Wave Adiabatic QC ideal for solving optimization problems
  - Scaling exponentially, from 50 to 2000 to 5000 quantum bits (qubits)
  - Quadratic Unconstrained Binary Optimization (QUBO) solver
  - Use software layers, or directly
- Already applied to Scheduling
  - Job Shop Scheduling: demonstrated at NASA Ames
  - However, the DSN type of scheduling problem has not yet been mapped to D-Wave

D-Wave



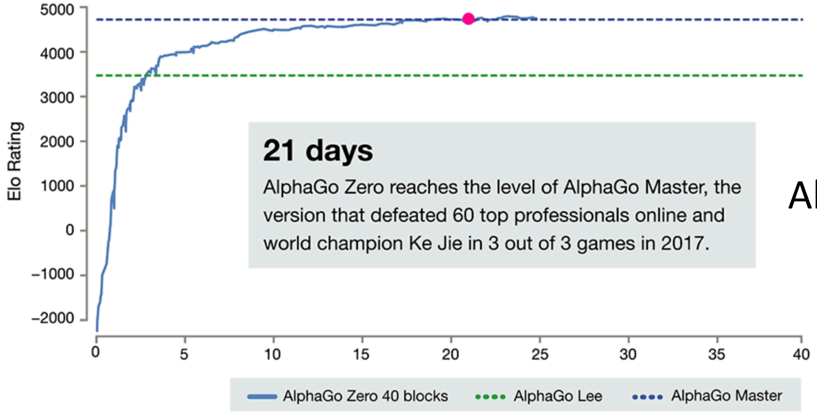




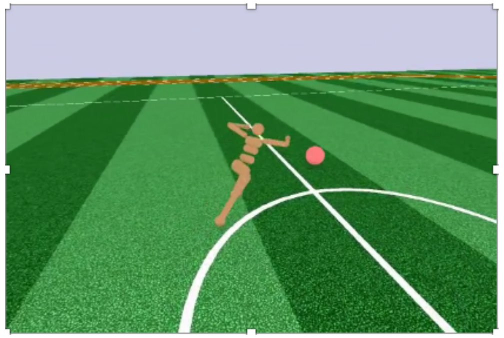
Deep Reinforcement Learning

- Deep Learning has revolutionized "AI" problems
    - Language translation
    - Text summarization, image captions
    - Object detection in images
    - Face recognition
    - Predicting 'action' in videos
- Deep RL has 'solved' many problems from scratch
    - Games: AlphaGo Zero plays Go better than humans after a week of training (self-play)
    - Locomotion: Robots learn to 'walk' without any innate programming
- Already applied to Scheduling
    - Microsoft DeepRM: Resource Mgmt.
    - Real-time Network & CPU Scheduling

Waymo

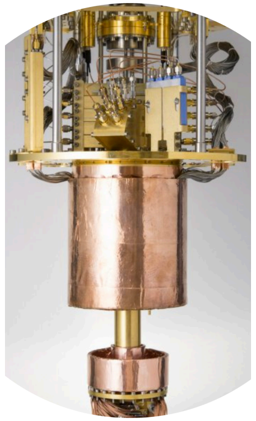


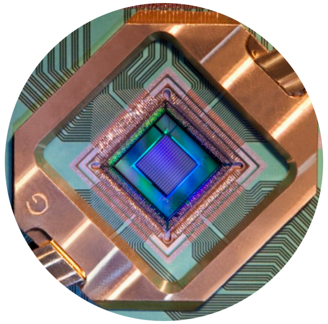
AlphaGoZero



Three Optimization Approaches

- Mixed Integer Linear Programming (MILP), as a baseline
  - Integer & Continuous Variables
  - Constraints
  - Quadratic Objective Function
- D-Wave: Quadratic Unconstrained Binary Optimization (QUBO)
  - Binary Variables
  - Quadratic Objective Function
- DeepRL – Agent ”learns” a Sequence of Actions
  - Binary Variables → Matrix Encoding for “state”
  - Constraints → Legal Actions for Agent
  - Reward Function → Agent learns a “policy”,
    a sequence of actions to deconflict the schedule



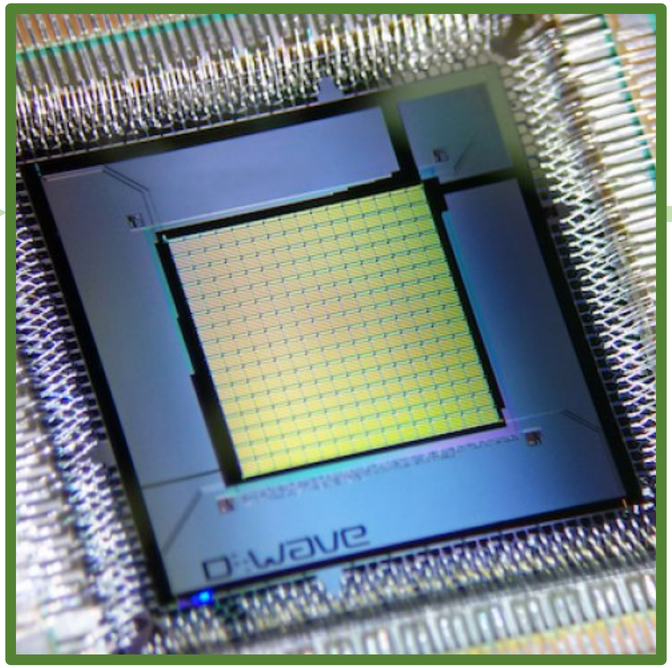


Quantum Computing

Quantum computing is the use of quantum-mechanical phenomena such as superposition and entanglement to perform computation.

Optimization with D-Wave quantum annealer
(NASA-Ames, D-Wave, Amazon Braket, LANL)

DSN requests →

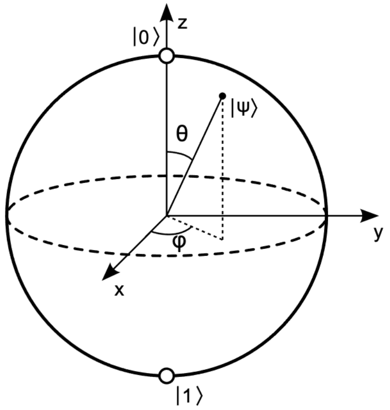


→ DSN schedules

Quantum Computing

Quantum computation and quantum information is the study of the information processing tasks that can be accomplished using quantum mechanical systems.

One qubit



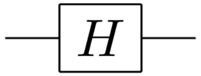
$$\frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

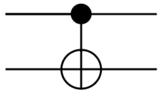
Multiples qubits

$$|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$$

Gates

Classical AND 

Hadamard 

CNOT 

Quantum parallelism

“simultaneous” evaluation.

Require specifically designed measurements to extract information.

Algorithms kinds:

-quantum version of the Fourier transform

-quantum search algorithm

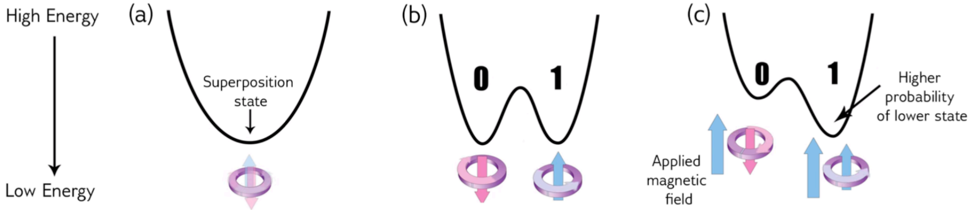
-quantum simulations

Message:

- Access to gigantic search space
- Single-shot like readout (~~speed~~)
- Require specifically designed algorithms to get the information
- Difficult to keep the quantumness on large scales

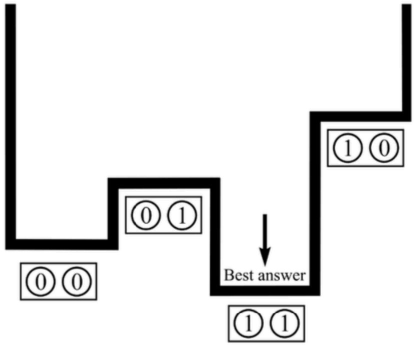
Quantum Annealing

Quantum annealing (QA) is a metaheuristic for finding the global minimum of a given objective function over a given set of candidate solutions (candidate states), by a process using quantum fluctuations.



D-Wave The Quantum Computing Company™ is a Canadian company founded in 1999.

It sells "the world's first commercially available quantum computer"

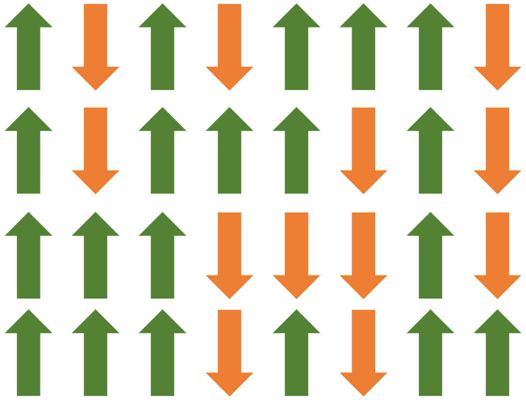


Description: A 2-dimensional array of magnetic moments $\vec{s}$.

What it does: solve a particular (Ising) optimization problem

$$E(\boldsymbol{s}|\boldsymbol{h},\boldsymbol{J}) = \left\{ \sum_{i=1}^{N} h_i s_i + \sum_{i<j}^{N} J_{i,j} s_i s_j \right\} \qquad s_i \in \{-1,+1\}$$

Provided $\boldsymbol{h}$ and $\boldsymbol{J}$ find the configurations of $\{\vec{s}\}$ that minimizes E.



Pros: It exists, it works, we have access to it. NASA Ames has 2038 qubits. D-Wave is working on the 5,640 Pegasus qubit.

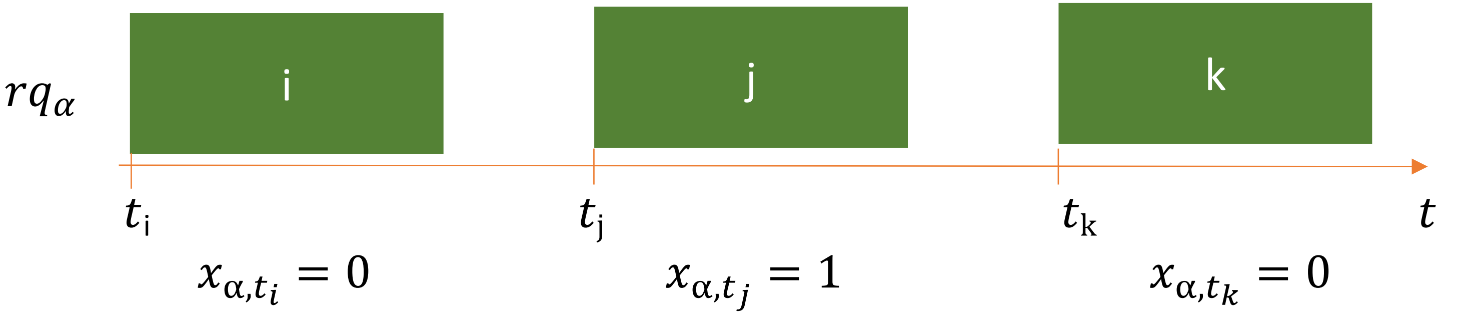
Cons: restricted purpose (akin to GPU or FPGA).

QUBO Formulation: Quadratic Unconstrained Binary Optimizaton

$$x_{i,t} = \begin{cases} 1: & \text{request } rq_i \text{ starts at time } t, \\ 0: & \text{otherwise.} \end{cases}$$

- **an operation must start once and only once:**

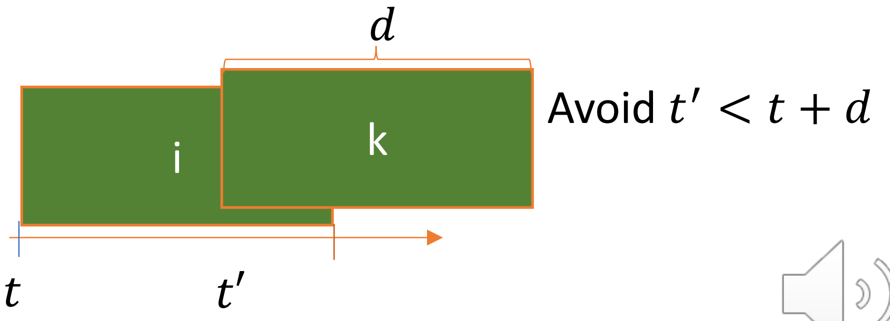
$$h_1(\bar{x}) = \sum_{i \in RQ} \left(\sum_{t \in T_i} x_{i,t} - 1 \right)^2$$



- **one job running on each equipment at any given point in time:**

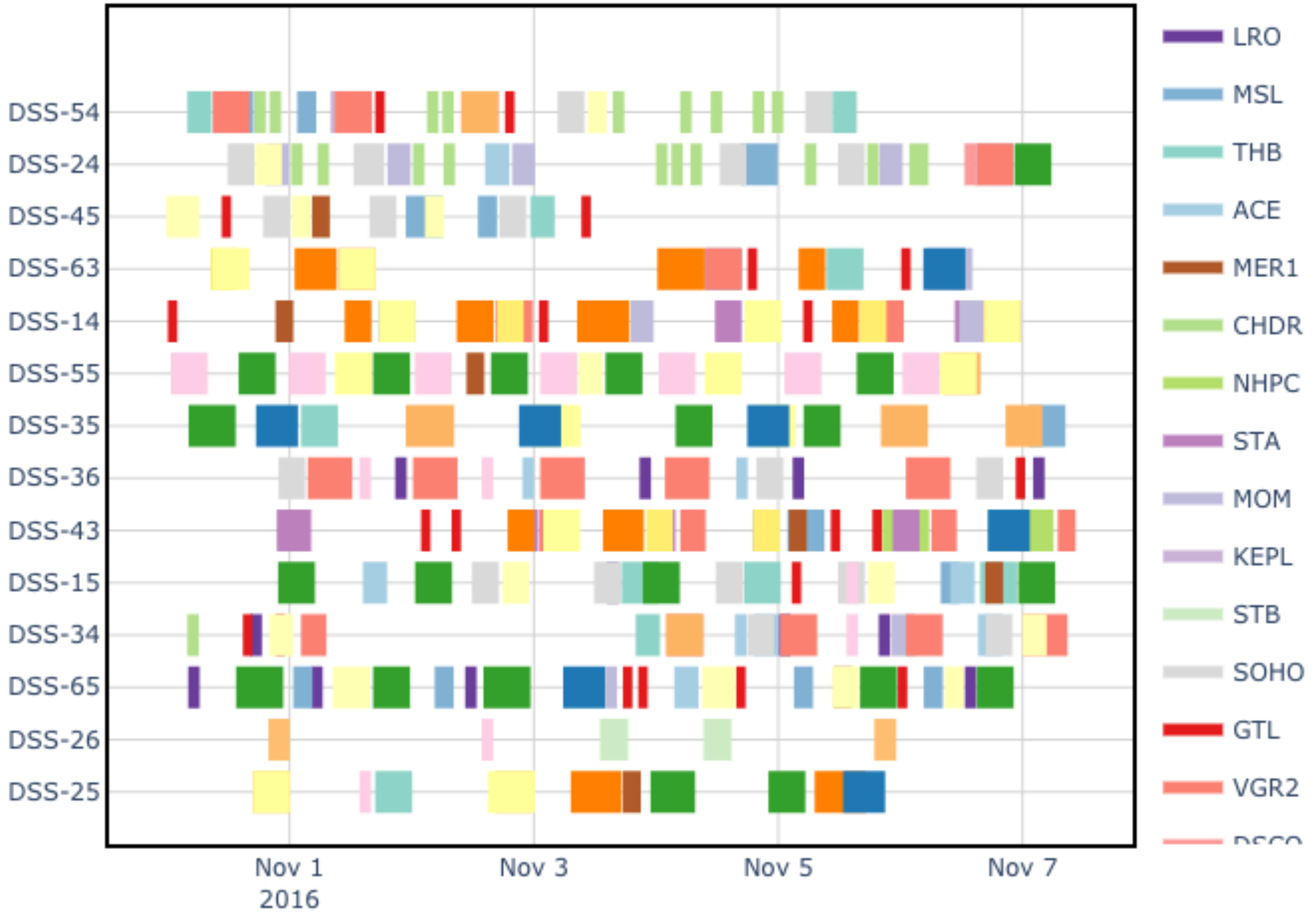
$$h_2(\bar{x}) = \sum_{(i,t,k,t') \in A} x_{i,t} x_{k,t'}$$

$x_{i,t}$	$x_{k,t'}$	$x_{i,t}x_{k,t'}$
0	0	0
0	1	0
1	0	0
1	1	1



One QUBO Formulation: multiple solvers

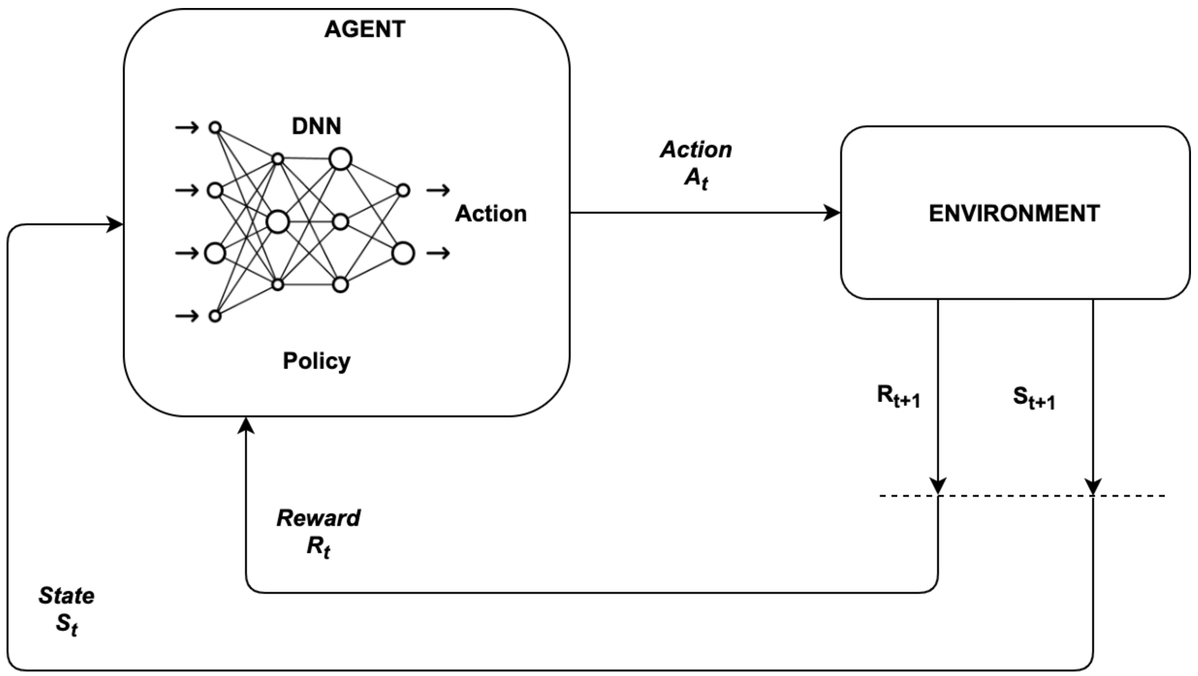
platforms	variables
\\ **Quantum** (annealer): D-Wave The Quantum Computing Company™, Amazon Braket	2,000
\\ **hybrid** (quantum/classical): D-Wave Leap	10,000
\\ **Quantum inspired optimization** (QIO): Meta-Analytics, Microsoft Azure	+100,000



DSN schedule run with Alpha-QUBO. The vast majority of the request were fulfilled (99%) but some conflicts remain. Some odd features are due to the DSN input data, e.g. the antenna DSS-45 was decommissioned half way through this particular week (44 of 2016).

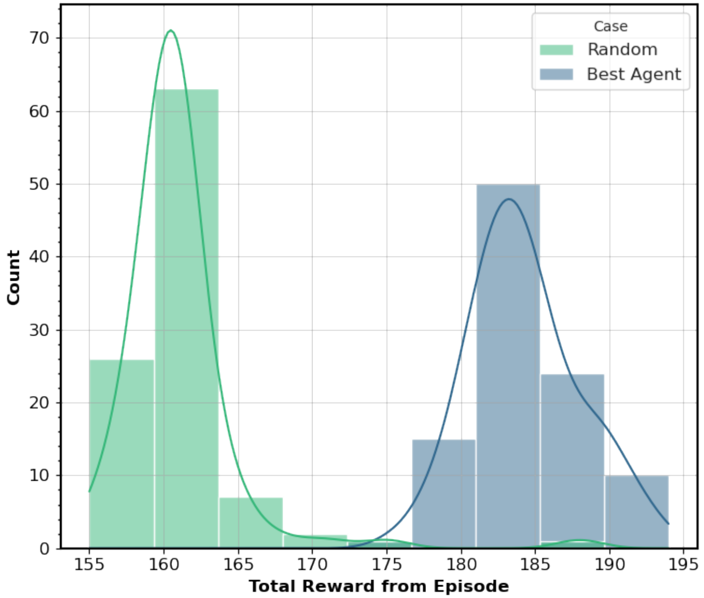
Deep Reinforcement Learning (now using RLlib)

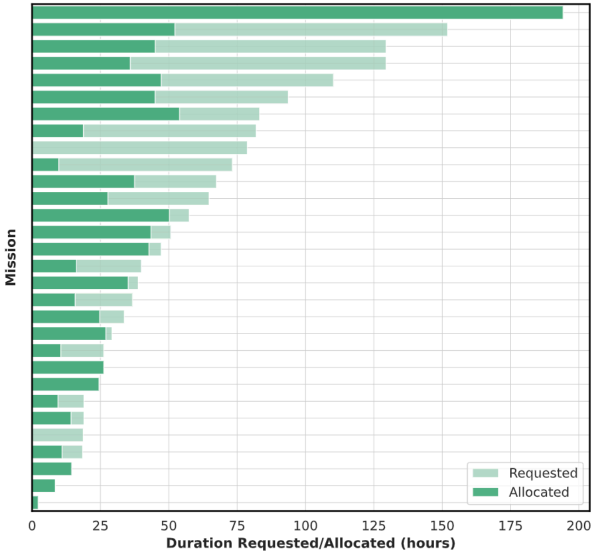
- Agent learns a sequence of "actions" to solve the problem (de-conflict the schedule).
  - Track actions: place, move, drop, re-select, etc.
- Agent encodes policy in a Deep Neural Net (DNN)
- Agent selects next track to place
- Agent queries the environment at current 'state' and receives a choice of legal actions
- Agent chooses an action, updates state, and receives an incremental reward
- Action space is large so 'learning' is difficult and requires many iterations
- Policy: Tradeoff between exploration & exploitation

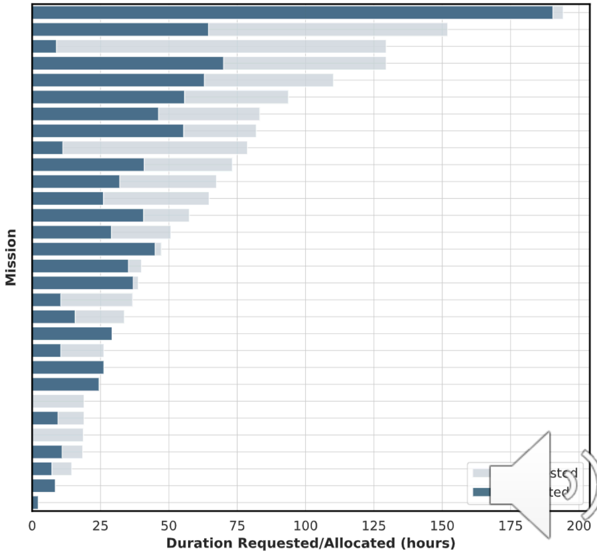


Early DeepRL Results

- Improved formulation using capabilities of RLlib
- Same problem: week 44 of 2016
- Agent beginning to learn (better than random)
- Distribution of total rewards for best agent versus random (upper right)
- However, scheduled hours per mission show no or only small increases (lower right).
- Number of scheduled requests low compared to QUBO or MILP
- Next steps:
    - Improved encoding of action space into 'features'
    - Dynamic action spaces
    - More experiments, longer training times







Mixed Integer Linear Programming (MILP v2)

- Variables are binary, integers, or continuous
- Maximize an objective subject linear constraints
- Large number of constraints to be handled
  - Some require adding variables
- Use convex relaxation to reduce the number of variables (relax integers to continuous variables)
- E.g. 200,000 variables reduced to 30,000
- Solve smaller problem using AMPL / Gurobi

Convex Relaxation

MILP Formulation: version 2 is complicated!

Parameters

$\boldsymbol{R} \in \{0,1\}^{\lvert\mathcal{R}\rvert \times \lvert\mathcal{V}\rvert}$	resource-viewperiod adjacency matrix
$\boldsymbol{A} \in \{0,1\}^{\lvert\mathcal{A}\rvert \times \lvert\mathcal{V}\rvert}$	activity-viewperiod adjacency matrix
$\boldsymbol{V} \in \{0,1\}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	viewperiod-time epoch adjacency matrix
$\boldsymbol{d}_{\min} \in \mathbb{Z}_+^{\lvert\mathcal{A}\rvert}$	minimum tracking duration vector
$\boldsymbol{d}_{\max} \in \mathbb{Z}_+^{\lvert\mathcal{A}\rvert}$	maximum tracking duration vector
$\boldsymbol{\delta}_{\uparrow} \in \mathbb{Z}_+^{\lvert\mathcal{A}\rvert}$	setup duration vector
$\boldsymbol{\delta}_{\downarrow} \in \mathbb{Z}_+^{\lvert\mathcal{A}\rvert}$	teardown duration vector
$\gamma_{\downarrow} \in \mathbb{N}^{\lvert\mathcal{V}\rvert}$	minimum downtime vector
$\gamma_{\uparrow} \in \mathbb{N}^{\lvert\mathcal{V}\rvert}$	minimum uptime vector

Integer Variables

$\boldsymbol{X} \in \{0,1\}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	viewperiod indicator
$\boldsymbol{a} \in \{0,1\}^{\lvert\mathcal{A}\rvert}$	activity assignment vector

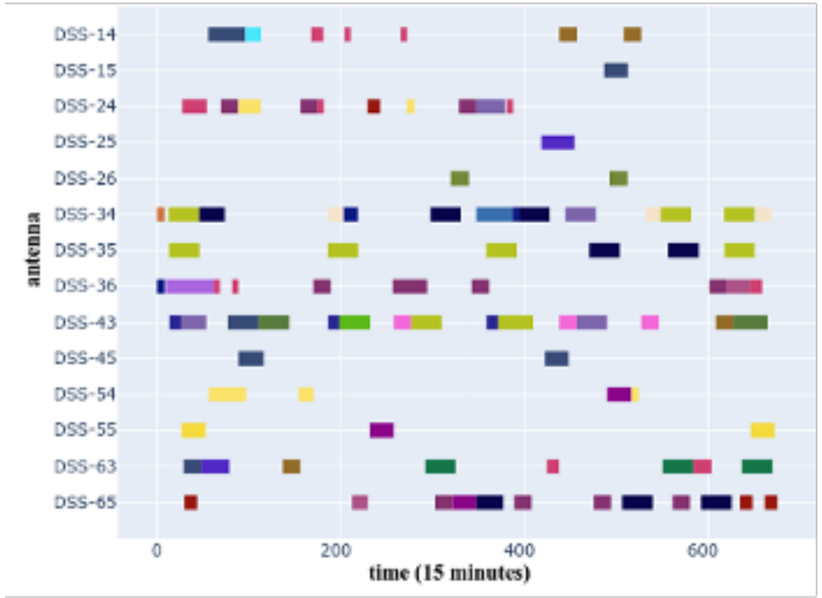
Continuous Variables

$\boldsymbol{X}^{\uparrow}, \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	startup indicator
$\boldsymbol{X}^{\downarrow} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	shutdown indicator
$\boldsymbol{U}, \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	setup indicator
$\boldsymbol{D} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}$	teardown indicator

$$
\begin{aligned}
\underset{\substack{\boldsymbol{X} \in \{0,1\}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} \\ \boldsymbol{a} \in \{0,1\}^{\lvert\mathcal{A}\rvert} \\ \boldsymbol{X}^{\uparrow},\, \boldsymbol{X}^{\downarrow} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} \\ \boldsymbol{U},\, \boldsymbol{D} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert}}}{\text{maximize}} \quad & \mathbf{1}_{\lvert\mathcal{A}\rvert}^{\top} \boldsymbol{a} && && \text{(1a)} \\
\text{subject to} \quad & \boldsymbol{X} \le \boldsymbol{V} && && \text{(1b)} \\
& \mathbf{0}_{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} \le \boldsymbol{X}^{\uparrow} \le \mathbf{1}_{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} && && \text{(1c)} \\
& \mathbf{0}_{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} \le \boldsymbol{X}^{\downarrow} \le \mathbf{1}_{\lvert\mathcal{V}\rvert \times \lvert\mathcal{T}\rvert} && && \text{(1d)} \\
& X_{v,t} - X_{v,t-1} = X_{v,t}^{\uparrow} - X_{v,t}^{\downarrow} && \forall v \in \mathcal{V},\ \forall t \in \mathcal{T} && \text{(1e)} \\
& \sum_{\tau = t - \gamma_v^{\uparrow} + 1}^{t} X_{v,\tau}^{\uparrow} \le X_{v,t} && \forall v \in \mathcal{V},\ \forall t \in \mathcal{T} && \text{(1f)} \\
& \sum_{\tau = t - \gamma_v^{\downarrow} + 1}^{t} X_{v,\tau}^{\downarrow} \le 1 - X_{v,t} && \forall v \in \mathcal{V},\ \forall t \in \mathcal{T} && \text{(1g)} \\
& U_{v,t} = \sum_{\tau = t+1}^{t + \delta_{A(v)}^{\uparrow}} X_{v,\tau}^{\uparrow} && \forall v \in \mathcal{V},\ \forall t \in \mathcal{T} && \text{(1h)} \\
& D_{v,t} = \sum_{\tau = t+1-\delta_{A(v)}^{\downarrow}}^{t} X_{v,\tau}^{\downarrow} && \forall v \in \mathcal{V},\ \forall t \in \mathcal{T} && \text{(1i)} \\
& \boldsymbol{R}(\boldsymbol{D} + \boldsymbol{X} + \boldsymbol{U}) \le \mathbf{1}_{\lvert\mathcal{R}\rvert \times \lvert\mathcal{T}\rvert} && && \text{(1j)} \\
& \operatorname{diag}\{\boldsymbol{d}_{\min}\}\boldsymbol{a} \le \boldsymbol{A}\boldsymbol{X}\mathbf{1}_{\lvert\mathcal{T}\rvert} \le \operatorname{diag}\{\boldsymbol{d}_{\max}\}\boldsymbol{a} && && \text{(1k)} \\
& \boldsymbol{A}\boldsymbol{X} \le \boldsymbol{a}\mathbf{1}_{\lvert\mathcal{T}\rvert}^{\top} && && \text{(1l)} \\
& \boldsymbol{A}\boldsymbol{U}\mathbf{1}_{\lvert\mathcal{T}\rvert} \ge \operatorname{diag}\{\boldsymbol{\delta}^{\uparrow}\}\boldsymbol{a} && && \text{(1m)} \\
& \boldsymbol{A}\boldsymbol{D}\mathbf{1}_{\lvert\mathcal{T}\rvert} \ge \operatorname{diag}\{\boldsymbol{\delta}^{\downarrow}\}\boldsymbol{a} && && \text{(1n)}
\end{aligned}
$$

MILP Results for week 44 of 2016 (2 formulations)

FY 19

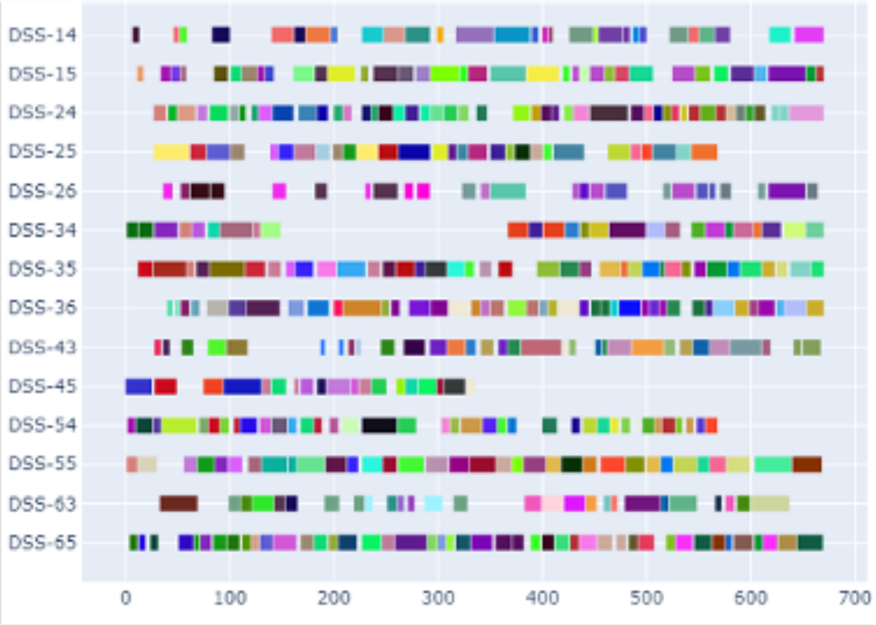


107/289

number of activities is
289
15 min time intervals

(w/ hyperparameter tuning, splitting, and weight randomization)

FY 20



264/289

Improvement in the generated schedule and number of requests fulfilled (264 out of 289, instead of 107) in moving to the second MILP formulation with convex relaxation.

Summary

- DSN antenna schedules are over-subscribed => hard optimization problem to be solved
  - Not fully formulated as a mathematical problem by mission schedulers
- Applying three techniques:
  - QUBO for Quantum Annealer or quantum-inspired solver
  - Deep Reinforcement Learning: large, challenging action space for agent
  - MILP formulation => many variables, need advanced techniques
- Have generated ‘good’ schedules from QUBO and MILP
- Mature and compare techniques in third year:
  - Metrics: unsatisfied time fraction per mission (hours), remaining conflicts, antenna utilization
  - Do schedules pass the “smell” test? Artifacts?

Publications

(Paper on quantum results in preparation.)

Goh, Edwin, Hamsa Venkataram, Mark Hoffmann, Mark Johnston, and Brian Wilson, “Scheduling the NASA Deep Space Network with Deep Reinforcement Learning,” submitted to *IEEE Aerospace Conference,* Yellowstone Conference Center, Big Sky, Montana, Mar 6 - Mar 13, 2021.

Sabol, Jonathon, Ryan Alimo, Farhad Kamangar, and Ramtin Madani, “Deep Space Network Scheduling via Mixed Integer Linear Programming”, submitted to *AIAA SciTech Conference*, virtual event, 11–15 January 2021.