Dynamic Spectrum Sharing: Lessons from the DARPA Spectrum Collaboration Challenge

John Shea

Team GatorWings

UNIVERSITY of FLORIDA

SPECTRUM COLLABORATION CHALLENGE

USING AI TO UNLOCK THE TRUE POTENTIAL OF THE RF SPECTRUM

SC2 Design Challenge–
Want Spectrum Management to:

- Respond in real time as users and traffic evolve:
  - Requires **autonomous intelligent** agents (take humans out of the loop)
  - Time scales of **seconds** (vs. one day in CBRS)
- Support diverse users and applications
  - Frequency channelization **not predetermined** (vs. 10 MHz channels in CBRS)
  - Need **scaffolding** to support information interchange: channels and protocols
  - Must **incentivize** accurate reporting and spectrum sharing
  - Greater **intelligence** needed => **more than traditional resource optimization**

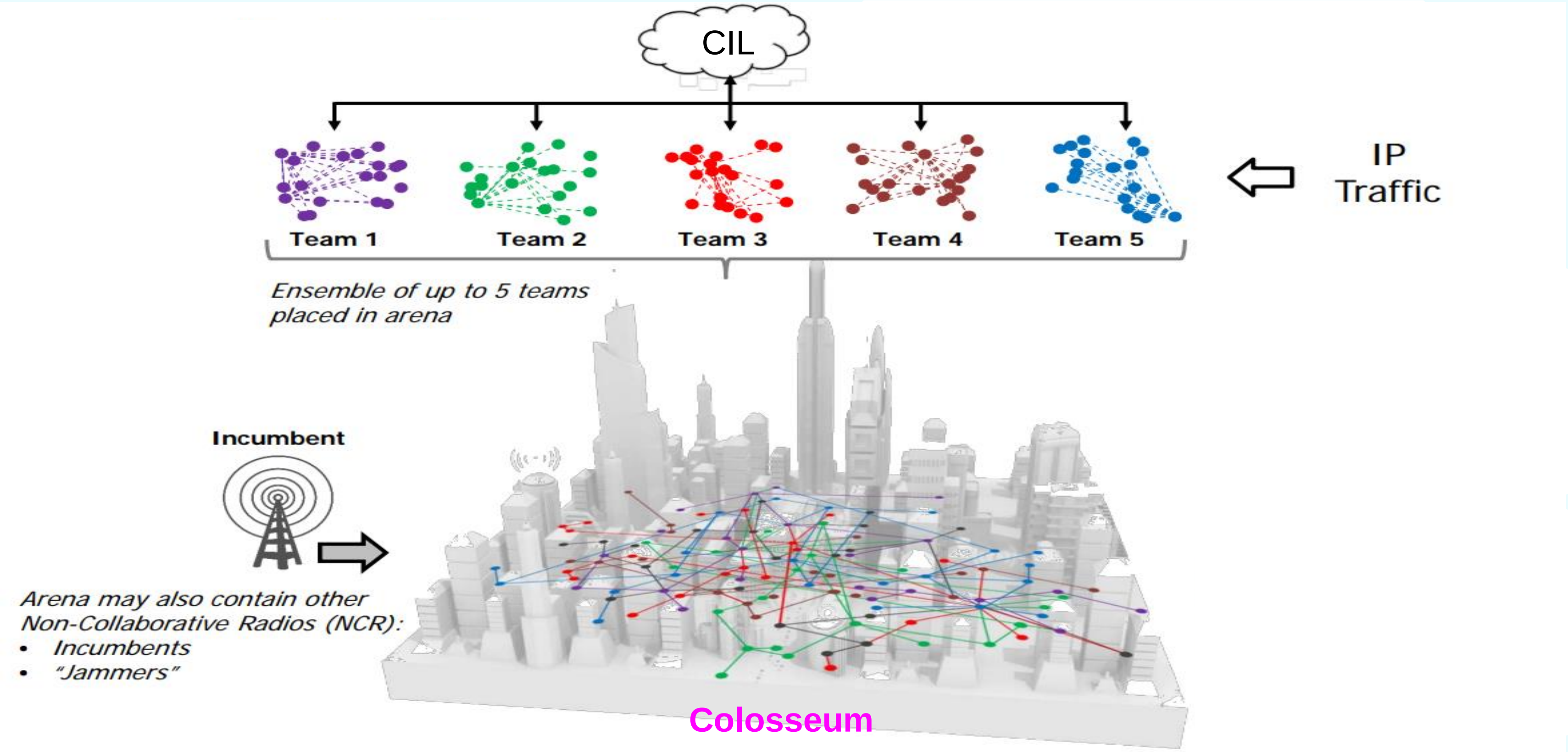
SC2 Design Challenge: Want Spectrum Management to

- Require minimal infrastructure and optimize spectrum usage across users' spatial distributions
  - **No central infrastructure** (i.e., SAS in CBRS)
  - Spectrum management should be **distributed** and **collaborative**
- Operate in presence of incumbents and interferers
  - Need **distributed sensing and reporting**

Conclusions from SC2:

- These goals can be achieved with existing technologies
- Much work to be done:
  - Efficiency?
  - Security / Privacy?
  - Incentivizing and enforcing?

Spectrum Sharing Scenario in SC2



SC2 Scoring Structure

- Each team scores points by delivering IP traffic flows achieving certain QoS mandates (throughput, latency, hold time, etc.)
- Team's match score $= \begin{cases} \text{min score} & \text{if min score} \leq \text{scoring threshold} \\ \text{achieved score} & \text{if min score} > \text{scoring threshold} \end{cases}$

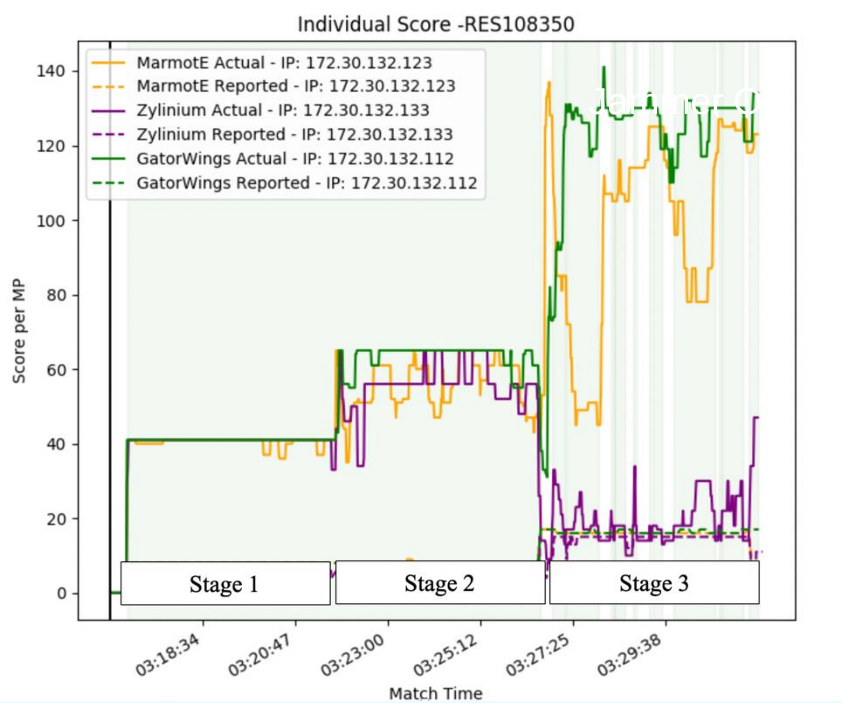
  where min score = minimum among all 5 teams' achieved scores
- A mixed cooperative/competitive game

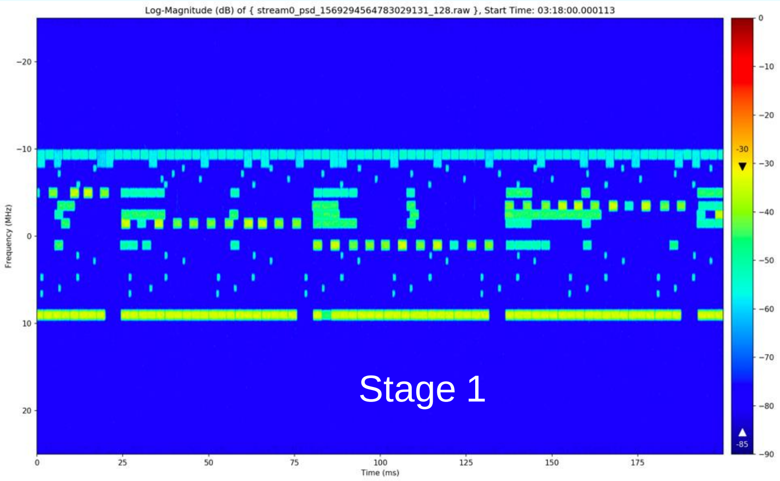
Information for Spectrum Sharing in SC2

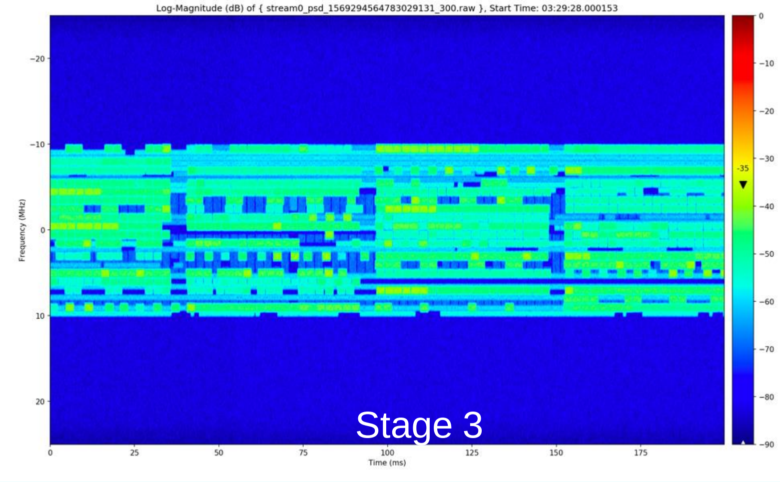
- Adapt strategy in presence of **rich but incomplete information**:
  - Do not know, but may learn, other teams' waveforms, protocols, strategies
  - No online scoring information, other than teams' estimates
  - Teams use CIL to report frequency use, radio locations, performance (score) estimates
    - Some CIL veracity checks on spectral use, scores
    - Teams do not have to report their true scores when their scores are above the threshold
  - Incumbents report channel usage, interference received and threshold, threshold violations
  - Spectrum sensing to estimate peer channel usage and detect jamming and active incumbents

Basic Dynamic Spectrum Sharing: Alleys of Austin

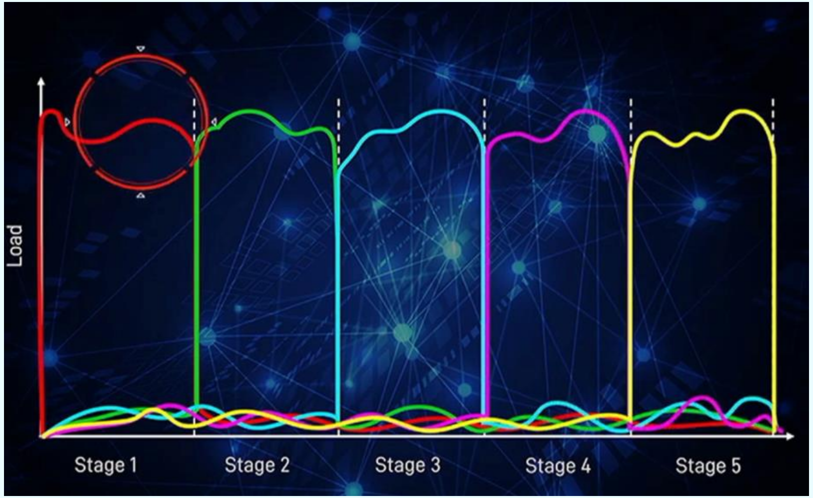
- 3 / 5 squads (networks) of solders sharing 20 MHz spectrum
- 3 stages of increasing IP traffic demands
    - Stage 1: VOIP & C2 streams
    - Stage 2: Stage 1 + video stream & file
    - Stage 3: Stage 2 + more video streams & files

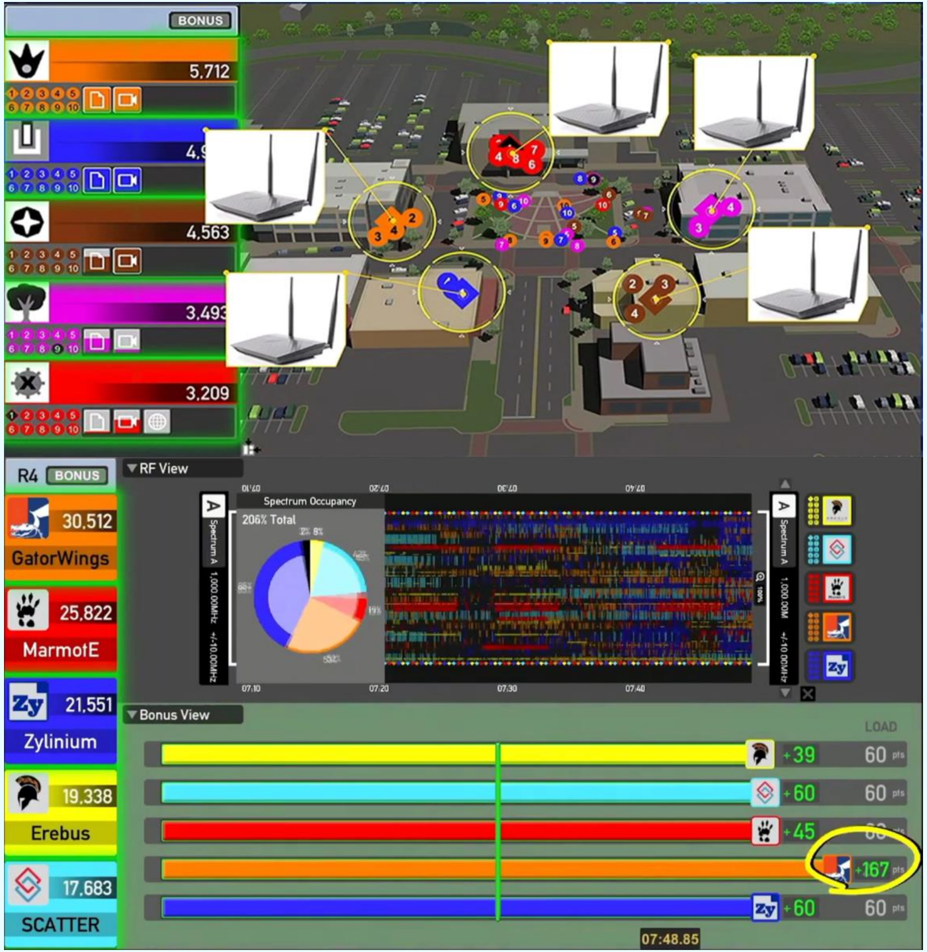

Individual Score -RES108350
MarmotE Actual - IP: 172.30.132.123
MarmotE Reported - IP: 172.30.132.123
Zylinium Actual - IP: 172.30.132.133
Zylinium Reported - IP: 172.30.132.133
GatorWings Actual - IP: 172.30.132.112
GatorWings Reported - IP: 172.30.132.112
Score per MP
0
20
40
60
80
100
120
140
Stage 1
Stage 2
Stage 3
03:18:34
03:20:47
03:23:00
03:25:12
03:27:25
03:29:38
Match Time


Log-Magnitude (dB) of { stream0_psd_1569294564783029131_128.raw }, Start Time: 03:18:00.000113
Frequency (MHz)
Time (ms)
Stage 1

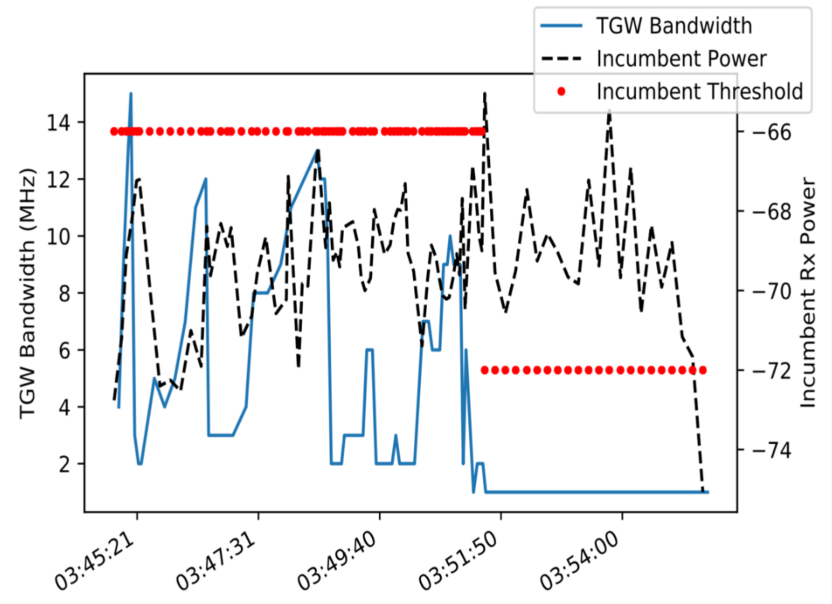

Log-Magnitude (dB) of { stream0_psd_1569294564783029131_300.raw }, Start Time: 03:29:28.000153
Frequency (MHz)
Time (ms)
Stage 3

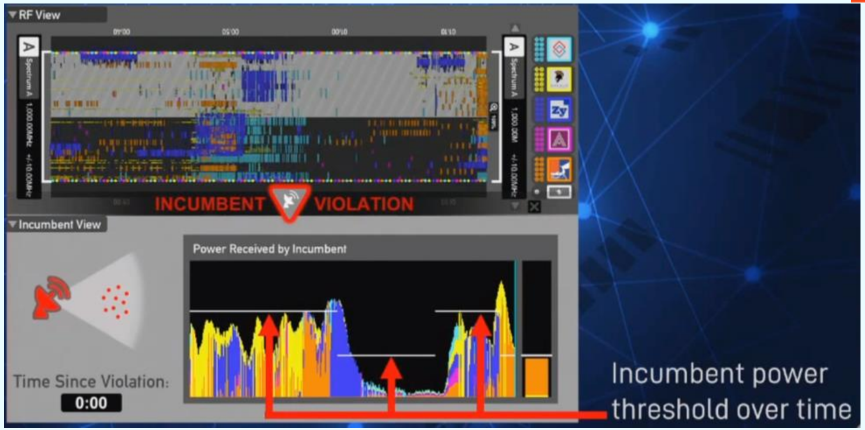
Traffic Surges (Slice of Life)

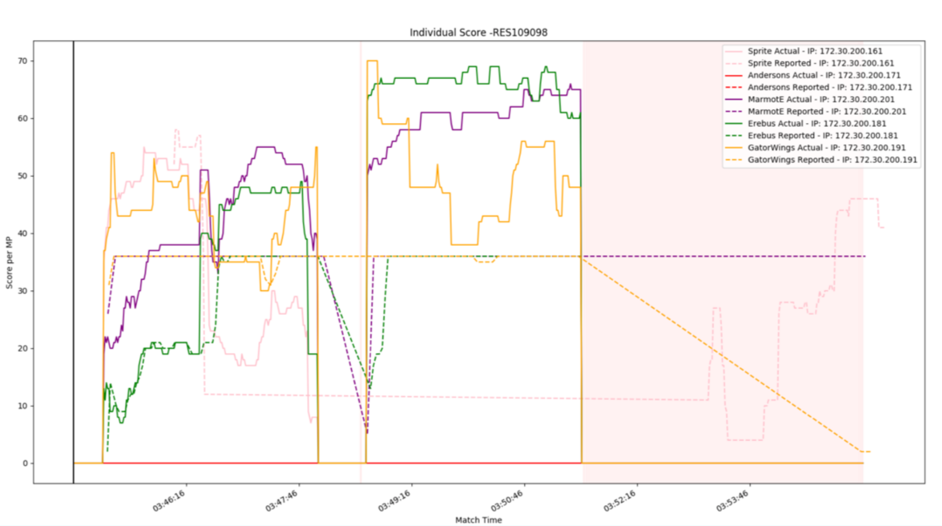




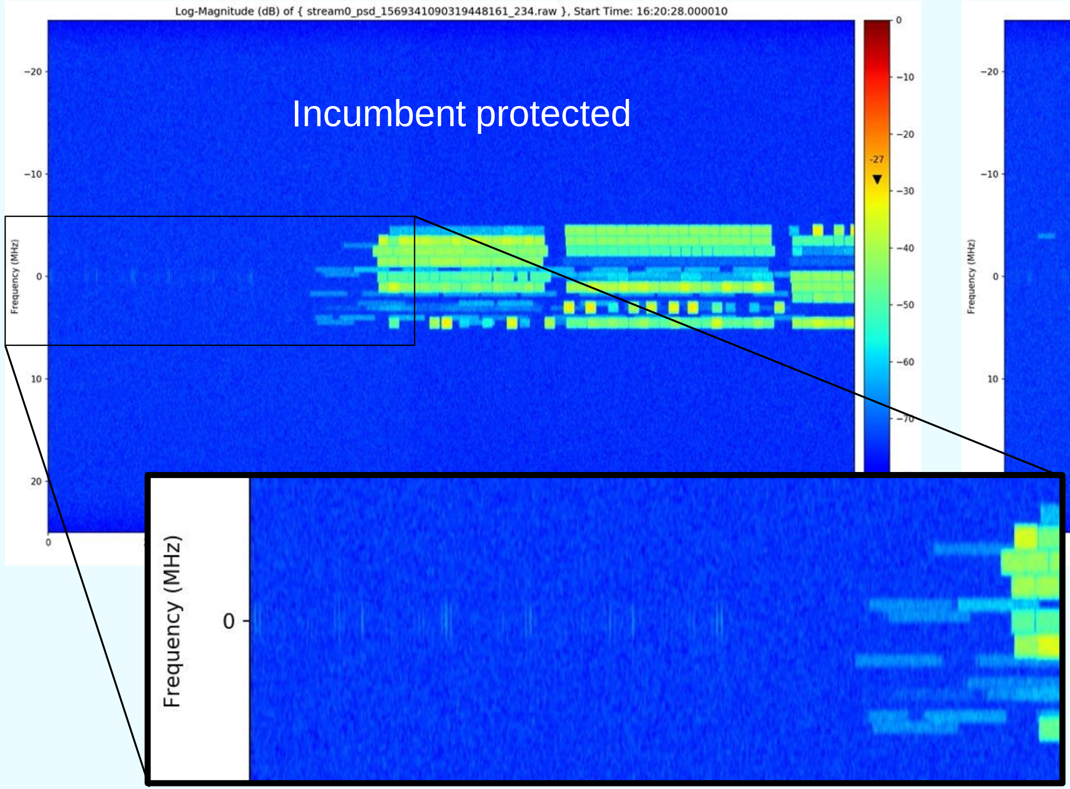
Passive Incumbent Protection

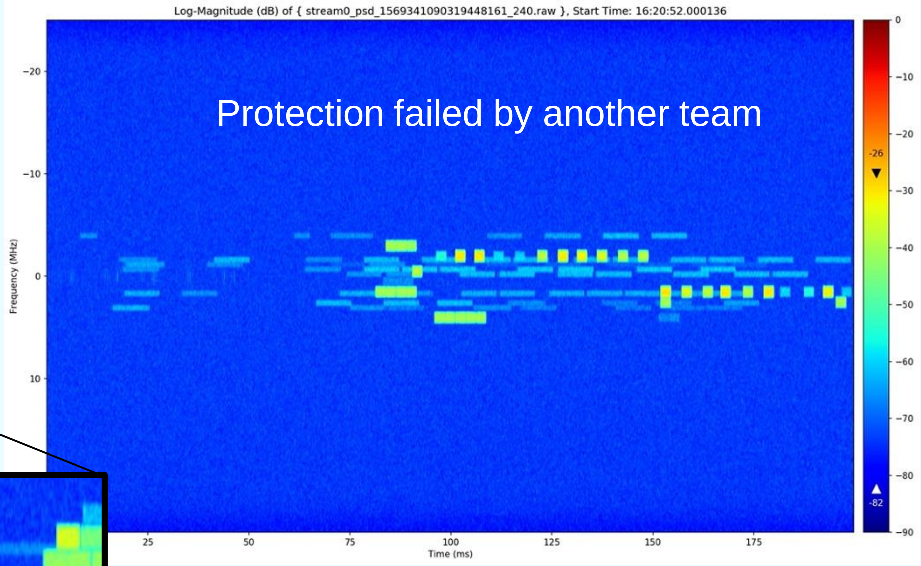


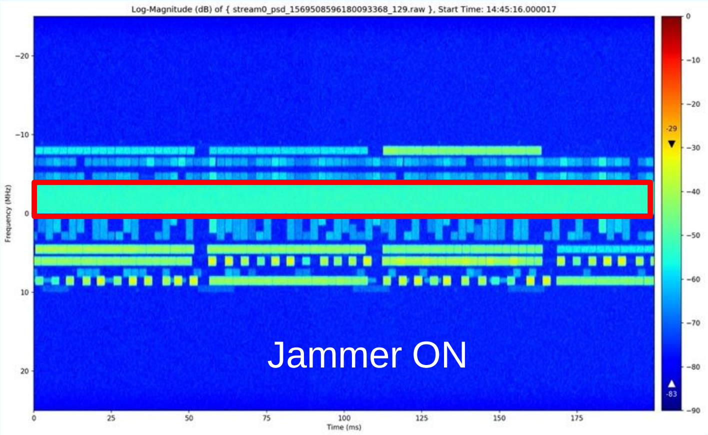


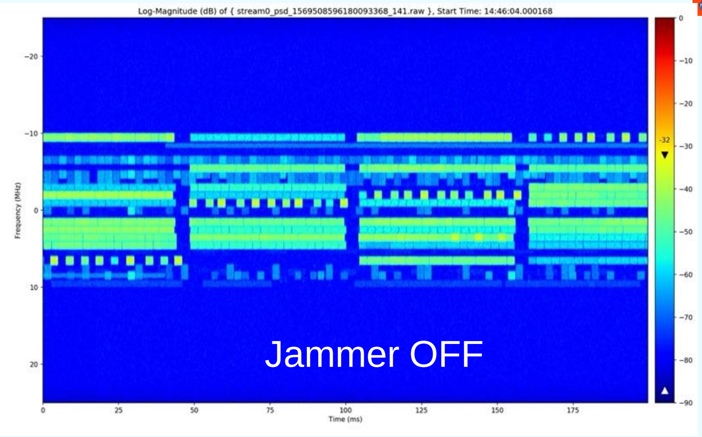


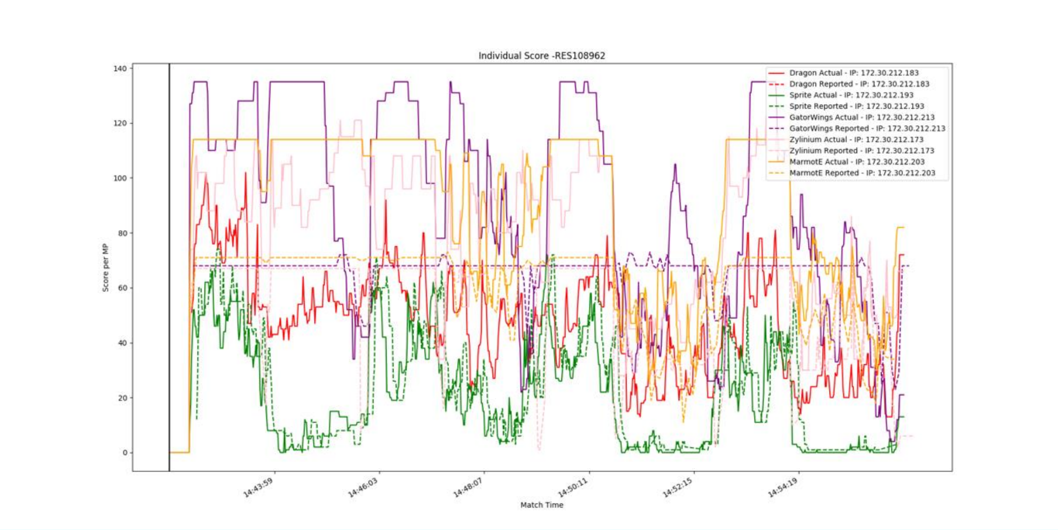
Active Incumbent (Vehicular Radar) Protection






Log-Magnitude (dB) of { stream0_psd_1569508596180093368_129.raw }, Start Time: 14:45:16.000017
Frequency (MHz)
Time (ms)
Jammer ON


Log-Magnitude (dB) of { stream0_psd_1569508596180093368_141.raw }, Start Time: 14:46:04.000168
Frequency (MHz)
Time (ms)
Jammer OFF


Individual Score -RES108962
Dragon Actual - IP: 172.30.212.183
Dragon Reported - IP: 172.30.212.183
Sprite Actual - IP: 172.30.212.193
Sprite Reported - IP: 172.30.212.193
GatorWings Actual - IP: 172.30.212.213
GatorWings Reported - IP: 172.30.212.213
Zylinium Actual - IP: 172.30.212.173
Zylinium Reported - IP: 172.30.212.173
MarmotE Actual - IP: 172.30.212.203
MarmotE Reported - IP: 172.30.212.203
Score per MP
Match Time

Our Strategy – Custom Radio Stack

- **Flexible/agile:** able to exploit opportunities in time, frequency & space (cooperative + competitive)
- **Robust:** adaptive and capable of operating in presence of strong interference (competitive)
- **Everything adaptive:**
  - PHY: Acquisition, Modulation, Coding, TX Power, RX Gain
  - LL: Channels and Time Slots/Channel, Mapping of SRCs to Time Slots
  - NET: Supported flows, admission control granularity down to individual files/bursts
  - Other: Channels to jam

Spectrum Sharing Decision Engine

- Decision engine attempts to maximize our team's match score :
  - **which flows are transmitted**
  - **which channels are used and by which radios**
  - **which flows are sent in which pockets (time-frequency resource unit)**
- Action space is huge!
  - 40 channels x 10 time slots = 400 pockets
  - As many as 100+ flows
  - 100^{400} possible pocket schedules!

Inputs to Decision Engine

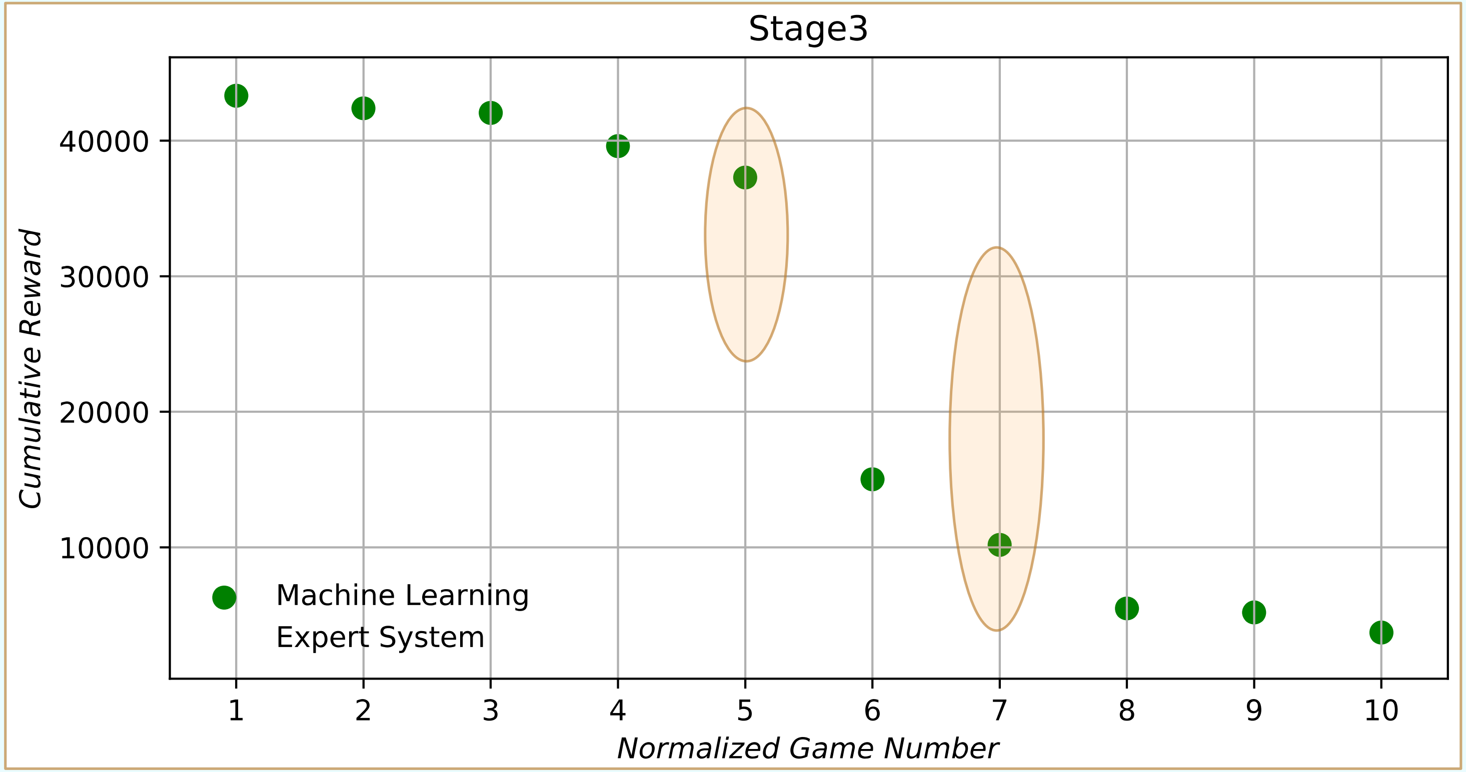
- Our team's **QoS Info and Performance**
  - QoS mandates for our team's flows
  - Estimated achieved mandates
- **Channel Info and Link Quality**
  - Estimated channel occupancies from our spectrum sensor (PSD measurements)
  - Channels used by our network and by peer networks
  - Computed SINRs from our interference map (GPS and voxel info from CIL messages)
  - Throughput per pocket expected between each SRC-DST pair
- **Peer IDs and Performance Info**
  - Inferred peer network IDs (based on CIL message characteristics)
  - Estimated achieved and total mandates (from CIL messages)

Decision Engine Design

- **No ML black box that can solve spectrum sharing problem**
- Decompose problem into smaller pieces:

1. **Channel selection**
    - Determines target set of channels *C* to be used by our network
    - ML and expert system/control/optimization approaches
2. **Admission control**
    - |*C*| determines number of pockets available
    - Estimates number of pockets needed to support each flow
    - Iterative process to determine set of flows to admit in order to maximize points scored
3. **Pocket schedule assignment**
    - Linear program to allocate number of pockets to satisfy latency requirements of all admitted flows
    - Greedy algorithm to assign pockets in each frame to satisfy mandates of all admitted flows
    - Maps to channels in *C* based on worst-case SINR over links of SRC-DST pairs in above assignments

Comparison of Expert System vs ML



Lessons Learned – Dynamic Spectrum Sharing

- A heterogeneous set of autonomous intelligent agents can manage spectrum in a distributed & collaborative fashion in time scales of seconds
- Dynamic sharing, traffic surge accommodation, passive and active incumbent protection can all be achieved with acceptable efficiency at the present time

- Essential to optimize strategy based on reward structure (scoring rules)

→ **Reward structure drives spectrum sharing behaviors**

Lessons Learned –
Machine Learning and AI for DSS

- No machine-learning black box that can solve spectrum sharing problem
- Domain-specific engineering to decompose problem
- Not enough training and validation data for ML
  - Need a less resource-intensive simulation environment to train ML algorithms
- Peer strategies (and probably radios) rapidly changed during last few weeks of SC2
  - ML algorithm (operation after training) couldn’t catch up
  - Switched to ES algorithm
  - Probably need more exploration and switching system to cope with rapid updates

Moving Forward – Technical Eco-system for DSS

- Suitable DSS system architecture (centralized, distributed, hybrid, geography, etc)
- Incentive frameworks to encourage desirable DSS behaviors
- Collaboration/information-sharing DSS protocols
- More efficient intelligent, autonomous agents to implement DSS strategies
- Radio & network design for DSS
- Privacy & security in DSS
- Monitoring and enforcement of DSS rules
- Performance metrics
- Test and experimentation
- Device/network certification

ML/AI Research Problems in DSS

- How to generate/collect suitable data sets for ML?
  - Simulation, RF emulation, or RF measurement?
- How to train multi-agent reinforcement learning agents with large state and action spaces?
- How to ensure ML solutions are robust to new situations?
- How can ML be used to police for noncompliant/abnormal behaviors?
- How to leverage ML in preserving privacy during exchange of collaboration information?
- Can ML adapt incentive structures over time to enhance overall network performance?

Acknowledgements

Tan Wong

Team Leader
Physical Layer
Sensing and
Spectrum
Mapping

David Greene

FPGA
Network
Layer
IDE & Tools

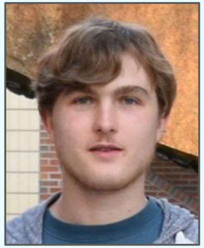
Tyler Ward

Physical
Layer
Link Layer
Control Plane

Marco Menendez

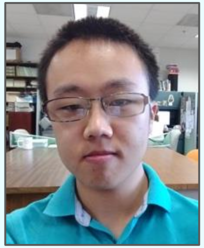
CIL
Development
Workflow
Optimization

Acknowledgements



Caleb Bowyer

Machine Learning



Shiming Deng[1]

Analysis & Visualization Tools

Quan Pham[2]

Channel Emulation

Josh Agarth[3]

CIL Development

[1]Phases 1 & 2: Now with L3Harris

[2]Phases 1 & 2: Now pursuing Ph.D.

[3]Phase 1: Now with Lockheed-Martin

- DARPA SC2 Team for running SC2 and developing Colosseum
- DARPA SC2 Prizes and NSF EAGER Grant 1738065 for supporting our team’s efforts