Incorporating learning in multi-agent systems such as task assignment for teams of limited-fuel unmanned aerial vehicles (UAVs) is challenging due to uncertainties in the assumed models and the large size of the planning space. Researchers have developed fast cooperative planners based on simple models (e.g., linear and deterministic dynamics), yet uncertainties in assumed models will impact the resulting performance. Learning techniques are capable of adapting the model and providing better policies asymptotically compared to cooperative planners, yet they often violate the safety conditions of the system due to their exploratory nature. Moreover they frequently require an impractically large number of interactions to perform well. We introduce the intelligent Cooperative Control Architecture (iCCA) as a framework for combining cooperative planners and reinforcement learning techniques. iCCA improves the policy of the cooperative planner, while reduces the risk and sample complexity of the learner.

### Contributions

- **Cooperative learners** formed by combining planners with RL algorithms through the intelligent Cooperative Control Architecture (iCCA)
- **Empirically showed the advantage** of our new approach over pure learning and pure planning methods for multi-agent missions with planning spaces up to 9 billion state-action pairs.
- Discussed the limitation of our approach and provided two potential solutions.

### Flow Chart

1. **Problem**
   - Real-world sequential decision making problems such as multi-agent domains have large state spaces and are sensitive to risky actions.

2. **Cooperative Planners:**
   - Fast and Safe
   - Sub-optimal

3. **Learning Methods:**
   - Asymptotically Optimal
   - Risky and Sample Inefficient

4. **How to achieve the best of the both worlds?** Realize safety, achieve optimality, and reduce sample complexity.

### Pedagogical Example

**UAV Mission Planning:** Two limited-fuel UAVs collaborate to obtain rewards on the graph marked as positive numbers. Probability of success is highlighted in the clouds. The timeframe for which the reward is available is highlighted in brackets. Planning spaces up to $9 \times 10^9$.

### Empirical Results

**Table:**

<table>
<thead>
<tr>
<th>Method</th>
<th>Optimality</th>
<th>Crash Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>SARSA</td>
<td>0.7716</td>
<td>0.0005</td>
</tr>
<tr>
<td>CBBA-conservative</td>
<td>0.7900</td>
<td>0.0221</td>
</tr>
<tr>
<td>CBBA-Aggressive</td>
<td>0.7900</td>
<td>0.0221</td>
</tr>
<tr>
<td>iCCA</td>
<td>0.7900</td>
<td>0.0221</td>
</tr>
<tr>
<td>AM-iCCA</td>
<td>0.7900</td>
<td>0.0221</td>
</tr>
</tbody>
</table>

**Analysis:**

- iCCA improved the best performance by 22% with respect to the allowed risk level of 20%.
- Regardless of how it is folded, AM-iCCA allows the learner’s policy to be optimally folded.
- The learned model guarantees the above might improve the performance of the model.

### Conclusion

- iCCA allows the fusion of reinforcement learning and cooperative planners. Resulting methods are shown to be more sample efficient than pure learning methods, while achieving better performance compared to pure planners in mission planning problems with sizes up to 9 billion state-action pairs. iCCA also mitigates the risk involved in learning methods.

---