Bahrami Parameter Sweep with REM Engine

This notebook demonstrates the Bahrami parameter sweep experiment using the REM (Retrieving Effectively from Memory) model.

Experiment Design:

- Fixed expert agent (A) with c = 0.7
- Sweep novice agent (B) ability from c = 0.1 to 0.9
- Test 5 group decision rules: CF, UW, DMC, DSS, BF
- Measure Collective Benefit Ratio = d'_team / max(d'_A, d'_B)

Key Question: Do groups perform better than their best individual member?

In [1]:

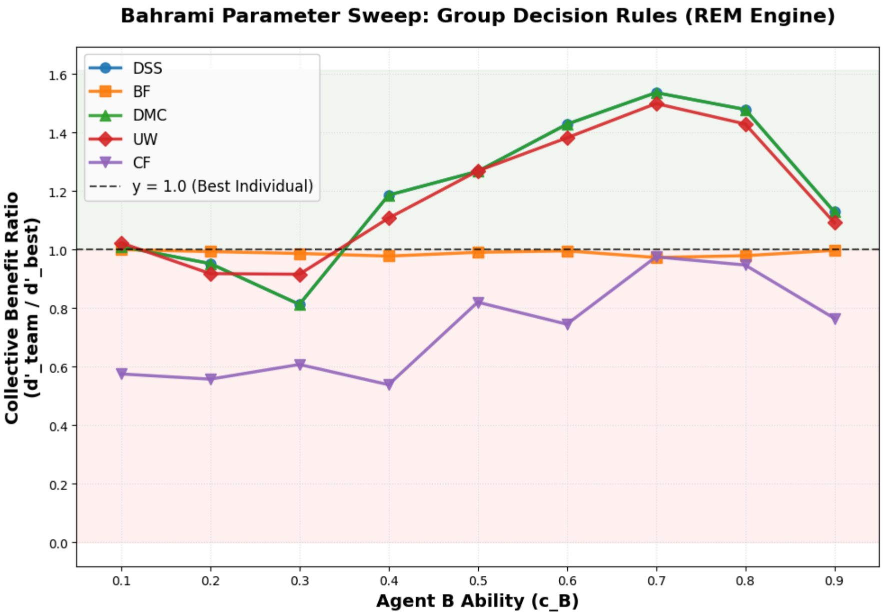
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from IPython.display import Image, display

import run_simulation
```

```
/Users/yiytan/opt/anaconda3/lib/python3.9/site-packages/pandas/core/computation/expressions.py:21: UserWarning: Pandas requires version '2.8.4' or newer of 'numexpr' (version '2.8.3' currently installed).
  from pandas.core.computation.check import NUMEXPR_INSTALLED
/Users/yiytan/opt/anaconda3/lib/python3.9/site-packages/pandas/core/arrays/masked.py:61: UserWarning: Pandas requires version '1.3.6' or newer of 'bottleneck' (version '1.3.5' currently installed).
  from pandas.core import (
```

In [2]:

```
# Run Bahrami parameter sweep
df = run_simulation.run_bahrami_sweep()
```

```
======================================================================
BAHRAMI PARAMETER SWEEP
======================================================================
Expert (A): c = 0.7
Novice (B): c ∈ [0.1, 0.9] (step = 0.1)
Trials per condition: 2000
Rules: CF, UW, DMC, DSS, BF
======================================================================

[c_B = 0.1]
  d'_A = 0.750, d'_B = 0.068, d'_best = 0.750
    CF: d'_team = 0.431, ratio = 0.575
    UW: d'_team = 0.767, ratio = 1.023
    DMC: d'_team = 0.758, ratio = 1.010
    DSS: d'_team = 0.758, ratio = 1.010
    BF: d'_team = 0.750, ratio = 1.000

[c_B = 0.2]
  d'_A = 0.841, d'_B = 0.021, d'_best = 0.841
    CF: d'_team = 0.469, ratio = 0.558
    UW: d'_team = 0.772, ratio = 0.918
    DMC: d'_team = 0.800, ratio = 0.952
    DSS: d'_team = 0.800, ratio = 0.952
    BF: d'_team = 0.835, ratio = 0.993

[c_B = 0.3]
  d'_A = 0.662, d'_B = 0.114, d'_best = 0.662
    CF: d'_team = 0.402, ratio = 0.608
    UW: d'_team = 0.606, ratio = 0.916
    DMC: d'_team = 0.538, ratio = 0.813
    DSS: d'_team = 0.538, ratio = 0.813
    BF: d'_team = 0.653, ratio = 0.987

[c_B = 0.4]
  d'_A = 0.474, d'_B = 0.105, d'_best = 0.474
    CF: d'_team = 0.255, ratio = 0.539
    UW: d'_team = 0.525, ratio = 1.107
    DMC: d'_team = 0.562, ratio = 1.186
    DSS: d'_team = 0.562, ratio = 1.186
    BF: d'_team = 0.464, ratio = 0.978

[c_B = 0.5]
  d'_A = 0.626, d'_B = 0.443, d'_best = 0.626
    CF: d'_team = 0.514, ratio = 0.820
    UW: d'_team = 0.794, ratio = 1.268
    DMC: d'_team = 0.794, ratio = 1.267
    DSS: d'_team = 0.794, ratio = 1.267
    BF: d'_team = 0.620, ratio = 0.991

[c_B = 0.6]
  d'_A = 0.358, d'_B = 0.583, d'_best = 0.583
    CF: d'_team = 0.434, ratio = 0.745
    UW: d'_team = 0.805, ratio = 1.381
    DMC: d'_team = 0.832, ratio = 1.428
    DSS: d'_team = 0.832, ratio = 1.428
    BF: d'_team = 0.580, ratio = 0.996

[c_B = 0.7]
  d'_A = 0.705, d'_B = 0.678, d'_best = 0.705
    CF: d'_team = 0.687, ratio = 0.975
    UW: d'_team = 1.056, ratio = 1.499
    DMC: d'_team = 1.082, ratio = 1.536
    DSS: d'_team = 1.082, ratio = 1.536
    BF: d'_team = 0.686, ratio = 0.973

[c_B = 0.8]
  d'_A = 0.746, d'_B = 0.657, d'_best = 0.746
    CF: d'_team = 0.706, ratio = 0.947
    UW: d'_team = 1.065, ratio = 1.429
    DMC: d'_team = 1.102, ratio = 1.478
    DSS: d'_team = 1.102, ratio = 1.478
    BF: d'_team = 0.730, ratio = 0.979

[c_B = 0.9]
  d'_A = 0.511, d'_B = 0.947, d'_best = 0.947
    CF: d'_team = 0.725, ratio = 0.765
    UW: d'_team = 1.034, ratio = 1.092
    DMC: d'_team = 1.069, ratio = 1.129
    DSS: d'_team = 1.069, ratio = 1.129
    BF: d'_team = 0.944, ratio = 0.997

✓ Results saved: bahrami_sweep_final.csv

✓ Plot saved: bahrami_sweep_plot.png

======================================================================
SUMMARY
======================================================================

Mean Collective Benefit Ratio by Rule:
rule
DMC    1.199879
DSS    1.199879
UW     1.181430
BF     0.988119
CF     0.725883
Name: collective_benefit_ratio, dtype: float64

======================================================================
✓ SIMULATION COMPLETE
======================================================================
```



Bahrami Parameter Sweep: Group Decision Rules (REM Engine)

```
In [3]: # Load and display results
        results = pd.read_csv('bahrami_sweep_final.csv')
        print("Results Summary:")
        print(f"Total rows: {len(results)}")
        print(f"\nColumns: {list(results.columns)}")
        print("\nFirst 10 rows:")
        results.head(10)
```

```
Results Summary:
Total rows: 45

Columns: ['c_A', 'c_B', 'rule', 'dprime_A', 'dprime_B', 'dprime_team', 'd_best', 'collective_benefit_ratio']

First 10 rows:
```

Out[3]:

	c_A	c_B	rule	dprime_A	dprime_B	dprime_team	d_best	collective_benefit_ratio
0	0.7	0.1	CF	0.749795	0.068107	0.431429	0.749795	0.575397
1	0.7	0.1	UW	0.749795	0.068107	0.766872	0.749795	1.022776
2	0.7	0.1	DMC	0.749795	0.068107	0.757590	0.749795	1.010397
3	0.7	0.1	DSS	0.749795	0.068107	0.757590	0.749795	1.010397
4	0.7	0.1	BF	0.749795	0.068107	0.749795	0.749795	1.000000
5	0.7	0.2	CF	0.840563	0.020713	0.468758	0.840563	0.557672
6	0.7	0.2	UW	0.840563	0.020713	0.771569	0.840563	0.917920
7	0.7	0.2	DMC	0.840563	0.020713	0.800379	0.840563	0.952194
8	0.7	0.2	DSS	0.840563	0.020713	0.800379	0.840563	0.952194
9	0.7	0.2	BF	0.840563	0.020713	0.834717	0.840563	0.993046

```
In [ ]: # Display the Bahrami plot
        display(Image('bahrami_sweep_plot.png'))
```