

World Cup Scheduling with Genetic Algorithm

By Rupam Tiwari and Utkarsha Pampatwar

Problem

Implement Genetic Algorithm (GA) to create a schedule of matches in world cup which follows certain constraints. Given, the number of teams, their names and locations, the algorithm creates a detailed schedule for the world cup. Each team plays every other team twice, once on their home ground and once on the opponent's home ground. Thus, if there are N teams, the number of matches will be $N*(N-1)$. Genetic algorithm is used to provide maximum optimization to reduce the number of violations occurred while creating the schedule. Below are some of the constraints taken into consideration.

- Scheduling a match on a location with bad weather on that day must be avoided
- No team will play games on consecutive dates
- Two games cannot happen on the same day
- Each location will host N-1 games
- Each Team should play exactly 2 games with all the other teams.
- Each team should have played $2*(N-1)$ games.

Implementation Design

Gene: Each gene corresponds to one match. The match contains information on the date, location, team1 and team2.

Chromosome: A list of matches corresponding to one schedule for the world cup. If number of teams is N, this list will have $N*(N-1)$ matches.

Population: A list of randomly generated schedules. The goal is to achieve the most optimized schedule from population having highest fitness score.

Fitness function: The fitness of a schedule is dependent on the number of violations that occurs during its creation. Violations are updated every time if any of the constraints are not satisfied. Constraint for weather is computed based on the weather index (Probability that the weather will be bad on that day at that location). Weather index greater than 70, will increase the conflict by 1. Since a high value of conflicts implies low fitness, the fitness is inversely proportional to the conflicts.

$$Fitness = \frac{1}{1 + conflicts}$$

Crossing Over: The crossover takes traits from both parents, in this case two schedules. The crossover is done by choosing the best fitting individuals to contribute with high probability. This is done by killing a part of the population which are not fit based on the culling ratio (0.5). Then choosing individuals from the rest of the population to contribute to the cross-over.

This is done by choosing 2 individuals randomly from the survived population, for example,

Schedule	A A1, A2, A3, A4, A5
Schedule	B B1, B2, B3, B4, B5

Then picking a random cross-over point (Say 3),

Schedule	A A1, A2, A3, A4, A5
Schedule	B B1, B2, B3, B4, B5

Then interchanging all the genes before this point to form the offspring;

Schedule Offspring-1 **B1, B2, B3**, A4, A5

Schedule Offspring-2 **A1, A2, A3**, B4, B5

Mutation: A temporary schedule is created with randomly created matches and the matches of the schedule to be mutated are replaced randomly by the corresponding matches of this temporary schedule. The number of replacements can be increased/ decreased by altering the value of mutation rate.

Schedule A A1, A2, A3, A4, A5

Temporary Schedule B B1, B2, B3, B4, B5

Traverse schedule A and replace with elements from B at random. For example, replace A2 with B2 and A4 with B4. (Swaps chosen randomly)

Mutated Schedule A A1, **B2**, A3, **B4**, A5

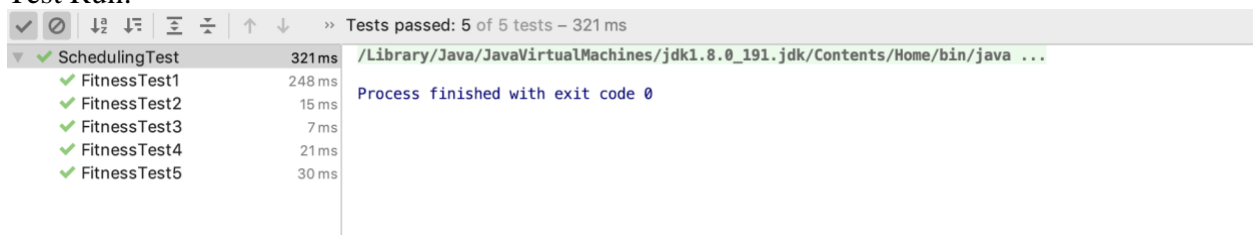
Evolution: The first step in evolution is to select the best individuals that are to be taken to the next generation. Two fittest individuals taken from samples of the population are crossed over to obtain a new individual. By taking individuals with better fitness, we make sure that the next generation has a better solution. Next, the elements of the individual are mutated to create the most optimal result.

Results

The test cases are written for testing if the fitness function and the algorithm is working correctly. In our case, the following cases are tested to validate our fitness function.

1. A schedule that satisfies all the criteria should produce 0 violations
2. Having multiple matches scheduled on a same day should generate a conflict for each violation
3. A team playing matches on consecutive days should increase the number of violation
4. Each violation of “2 matches with each opponent” contract should increase the number of violations
5. Location hosting more than allowed games should create a violation
6. Games scheduled on a bad weather (index > 70)
7. Each Team playing exactly $2*(N - 1)$ games, where N is the number of teams

Test Run:



The screenshot shows a test run window with a toolbar at the top containing icons for running, debugging, and other actions. The status bar indicates "Tests passed: 5 of 5 tests - 321 ms". The test results are listed in a table:

Test Name	Duration
✓ SchedulingTest	321ms
✓ FitnessTest1	248 ms
✓ FitnessTest2	15 ms
✓ FitnessTest3	7 ms
✓ FitnessTest4	21 ms
✓ FitnessTest5	30 ms

Below the table, the text "Process finished with exit code 0" is displayed.

Application Run Output:

```
/Library/Java/JavaVirtualMachines/jdk1.8.0_191.jdk/Contents/Home/bin/java ...
0 [main] INFO scheduling.Driver - Running algorithm with following configuration:
2 [main] INFO scheduling.Driver - scheduling.Population Size: 5000
2 [main] INFO scheduling.Driver - Colony Size: 100
2 [main] INFO scheduling.Driver - Tournament Selection Size: 25
4 [main] INFO scheduling.Driver - Mutation Rate: 0.02
4 [main] INFO scheduling.Driver - Crossover Rate: 0.75
4 [main] INFO scheduling.Driver - Culling Rate: 0.5

4108 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Generation - 1
4108 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Best Schedule:
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.07692
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 12
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Worst Schedule:
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.03226
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 30
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver -

Best Schedule:
4110 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Date | Team1 | Team2 | Location
*****
4112 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/22/2019 | SouthAfrica | EnglandTeam | SouthAfrica
4113 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/22/2019 | EnglandTeam | India | England
4113 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/23/2019 | India | Bangladesh | India
4113 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/23/2019 | India | SouthAfrica | India
4114 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/23/2019 | Bangladesh | EnglandTeam | Bangladesh
4114 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/26/2019 | Bangladesh | SouthAfrica | Bangladesh
4114 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/28/2019 | EnglandTeam | Bangladesh | England
4115 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/30/2019 | India | Bangladesh | India
4115 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/01/2019 | EnglandTeam | SouthAfrica | England
4115 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/05/2019 | SouthAfrica | India | SouthAfrica
4115 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/05/2019 | Bangladesh | EnglandTeam | Bangladesh
4116 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/06/2019 | SouthAfrica | India | SouthAfrica

6158 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Generation - 2
6158 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Best Schedule:
6158 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.12500
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 7
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Worst Schedule:
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.03448
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 28
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver -

Best Schedule:
6159 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Date | Team1 | Team2 | Location
*****
6160 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/20/2019 | SouthAfrica | Bangladesh | SouthAfrica
6160 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/20/2019 | SouthAfrica | India | SouthAfrica
6160 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/21/2019 | India | Bangladesh | India
6160 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/23/2019 | SouthAfrica | EnglandTeam | SouthAfrica
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/24/2019 | EnglandTeam | SouthAfrica | England
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/25/2019 | Bangladesh | India | Bangladesh
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/28/2019 | Bangladesh | SouthAfrica | Bangladesh
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/29/2019 | EnglandTeam | Bangladesh | England
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/01/2019 | India | SouthAfrica | India
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/01/2019 | Bangladesh | EnglandTeam | Bangladesh
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/02/2019 | EnglandTeam | India | England
6161 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/04/2019 | India | EnglandTeam | India

10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Generation - 8
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Best Schedule:
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.14286
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 6
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Worst Schedule:
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.05263
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 18
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver -

Best Schedule:
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Date | Team1 | Team2 | Location
*****
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/20/2019 | SouthAfrica | Bangladesh | SouthAfrica
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/20/2019 | SouthAfrica | India | SouthAfrica
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/23/2019 | SouthAfrica | EnglandTeam | SouthAfrica
10467 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/24/2019 | EnglandTeam | SouthAfrica | England
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/25/2019 | Bangladesh | India | Bangladesh
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/25/2019 | India | Bangladesh | India
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/28/2019 | Bangladesh | SouthAfrica | Bangladesh
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 04/29/2019 | EnglandTeam | Bangladesh | England
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/01/2019 | India | SouthAfrica | India
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/01/2019 | Bangladesh | EnglandTeam | Bangladesh
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/02/2019 | EnglandTeam | India | England
10468 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - 05/04/2019 | India | EnglandTeam | India
```


Best Schedule:

```
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Generation = 57
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Best Schedule:
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.50000
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 1
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Worst Schedule:
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.06250
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 15
32178 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver -
```

Best Schedule:

```

65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Generation - 151
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Best Schedule:
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 1.00000
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 8
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Worst Schedule:
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Fitness = 0.87692
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver - Violations = 12
65721 [ForkJoinPool.commonPool-worker-13] INFO scheduling.Driver -

```

Best Schedule:

```
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Generation - 1
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Best Schedule:
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Fitness = 0.08333
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Violations = 11
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Worst Schedule:
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Fitness = 0.83571
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver - Violations = 27
67796 [ForkJoinPool.commonPool-worker-18] INFO scheduling.Driver -
```

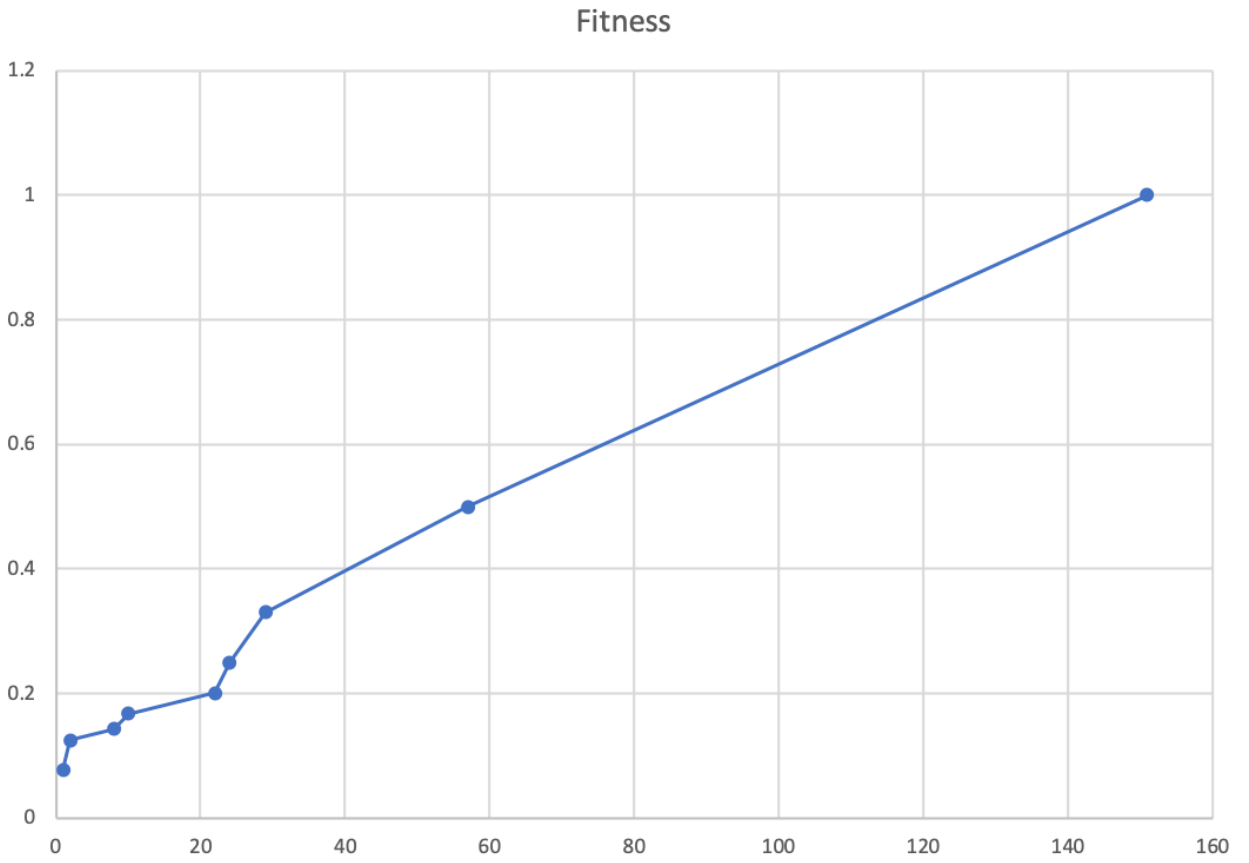
Some Converged solution stats:
Solution1:

Generation	Fitness	Violations
1	0.077	12
2	.125	7
8	.143	6

10	.167	5
22	.20	4
24	.25	3
29	.33	2
57	.5	1
151	1	0

Solution2:

Generation	Fitness	Violations
1	0.07	13
3	.10	9
4	.125	7
10	.167	5
14	.20	4
19	.25	3
40	.33	2
47	.5	1
83	1	0



Conclusion:

Since the implemented algorithm is trying to select the best species for next generation, with each iteration fitness is increasing. With above graph between generation iteration and fitness, it is pretty clear that the relation is linear. Which further testifies the fact of improvement in fitness score and adaptation of strongest. Though depending on parameters such as mutation rate solution might not converge for each experiment.