# **Complexity analysis of Perfect and Individually Rational Stability Concepts:**

Here we consider hedonic games in which each agent considers other agents as either friends, enemies or neutral agents. We provide algorithms to establish complexity analysis of perfect and individually rational stability concepts.

Let A denote set of players {1, 2, 3, ……………, n}

Let Ai denote set of coalitions in which player i is present

Let Δ~i+0- denote preferences of player i over coalition set Ai{\displaystyle \preceq }

Comparing two coalitions:

A coalition X is preferred to a coalition Y by player i if, firstly, for each friend fY of i in Y there is a friend fX of i in X that is at least as preferred by i as fY (i.e., fX Δ~i fY) and, secondly, for each enemy eX of i in X there is an enemy eY of i in Y that is at least as disliked by i as eX (i.e.,eX Δ~i eY). Additionally, these friends in X and enemies in Y both must be chosen pair wise distinctly. Thus, this conditional so implies that there must be at least as many of i’s friends in X as in Y and at most as many enemies of i in X as in Y. Clearly, comparing these two coalitions can be done in polynomial time by sorting the friends lists in the descending order and comparing them one by one followed by sorting the enemies lists in ascending order and comparing them one by one.

Stability Concepts:

* Perfect Coalition Structure: if each player weakly prefers her assigned coalition to every other coalition containing her.
* Individually Rational Coalition Structure: if every player weakly prefers her assigned coalition to being alone.

## **Verification Problems:**

Possible-Verification for Perfect Coalition Structure:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- > and coalition structure Γ

Output: Returns ‘YES’ if the given Γ is a possible perfect coalition structure, else returns ‘NO’

Algorithm1:

1. For all preference profiles(>) ∈ Xi = 1 to n Ext(>~i+0-)
2. Flag = ‘YES’
3. For i ∈ A
4. For C ∈ Ai
5. If C >i Γ(i)
6. Flag = ‘NO’
7. If Flag != ‘NO’ return ‘YES’
8. Return ‘NO’

Possible-Verification for Individually Rational Coalition Structure:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- > and coalition structure Γ

Output: Returns ‘YES’ if the given Γ is a possible individually rational coalition structure, else returns ‘NO’

Algorithm2:

1. For all preference profiles(>) ∈ Xi = 1 to n Ext(>~i+0-)
2. Flag = ‘YES’
3. For i ∈ A
4. {i} >i Γ(i)
5. Flag = ‘NO’
6. If Flag != ‘NO’ return ‘YES’
7. Return ‘NO’

Necessary-Verification for Perfect Coalition Structure:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- > and coalition structure Γ

Output: Returns ‘YES’ if the given Γ is a necessary perfect coalition structure, else returns ‘NO’

Algorithm3:

1. For all preference profiles(>) ∈ Xi = 1 to n Ext(>~i+0-)
2. For i ∈ A
3. For C ∈ Ai
4. If C >i Γ(i)
5. Return ‘NO’
6. Return ‘YES’

Necessary-Verification for Individually Rational Coalition Structure:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- > and coalition structure Γ

Output: Returns ‘YES’ if the given Γ is a necessary individually rational coalition structure, else returns ‘NO’

Algorithm4:

1. For all preference profiles (>) ∈ Xi = 1 to n Ext(>~i+0-)
2. For i ∈ A
3. {i} >i Γ(i)
4. Return ‘NO’
5. Return ‘YES’

Clearly all the above algorithms (Algorithm1 to Algorithm4) run in polynomial time as they involve only nested for loops.

Existence Problems:

Perfect Coalition Structure:

Algorithm5:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- >, > = Xi ∈ [A] >~i

Output: Returns ‘YES’ if there exists a coalition structure that is perfect for given preference profile, else returns ‘NO’

1. For i ∈ A
2. Si = Max(Ai) (Si denotes the coalition in Ai with maximum preference)
3. Initialize coalition structure Γ = NULL
4. For i ∈ A
5. Check in Γ if i is already in some coalition i.e. if Γ(i) != NULL
6. If Γ(i) == NULL
7. Γ(i) = Si
8. Else
9. If Γ(i) != Si
10. Return ‘NO’
11. Return ‘YES’

Run the above algorithm for all profiles in Ext(>~i+0-). If at least one of them returns ‘YES’ then possible existence for perfect coalition structure is ‘YES’. If all of them ‘YES’, then necessary existence for perfect coalition structure is ‘YES’. Else necessary existence is ‘NO’. if none of them returns ‘YES’ then possible existence is also ‘NO’.

Comparing two coalitions takes polynomial time. And hence finding max (Si) takes polynomial time. Checking if Γ contains coalition containing i also takes polynomial computing time. These two are nested inside a for loop. Therefore, the entire algorithm runs in polynomial time. Running the algorithm for all the profiles in Ext adds one more for loop on the outside which again falls under P.

Individually Rational Coalition Structure:

Algorithm6:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- >

Output: Returns |A| number of sets of coalitions in which each player has coalitions that have higher or equal preference to coalition in which he/she is alone

1. P1, P2, ……….., Pn
2. For i ∈ A
3. Pi = NULL
4. For x ∈ Ai
5. If x >~i {i}
6. Pi.append(x)

Now Pi denotes set coalitions containing i that are weakly preferred to {i}.

Algorithm7:

Input: A, < Δ~1+0-, Δ~2+0-, …………………, Δ~n+0- >, P = Xi ∈ [|A|] Pi

Output: Returns ‘YES’ if there exists a coalition structure that is individually rational for given preference profile, else returns ‘NO’

1. For i ∈ A
2. Si = Max (Pi) (Si denotes the coalition in Ai with maximum preference)
3. Initialize coalition structure Γ = NULL
4. For i ∈ A
5. Check in Γ if i is already in some coalition i.e. if Γ(i) != NULL
6. If Γ(i) == NULL
7. Γ(i) = Si
8. Else
9. If Γ(i) != Si
10. Return ‘NO’
11. Return ‘YES’

If the above algorithm returns ‘YES’ for some profile, then there exists a stable coalition structure for that profile in which every player i chooses to stay in the assigned coalition than to move to {i}. If it returns ‘NO’, then it means that there is no coalition structure in which every player prefers to stay in his assigned coalition to being alone. So, the above algorithm checks for existence of individually rational coalition structure.

Now, run the above algorithm for all profiles in Ext(>~i+0-). If it returns ‘YES’ for all profiles, then necessary existence of individually rational coalition structure is ‘YES’. If it returns ‘YES’ for some profiles, then possible existence of individually rational structure is ‘YES’ and necessary existence is ‘NO’. If it returns ‘NO’ for all profiles, then both necessary and possible existence of individually rational structure is ‘NO’.

Running Algorithm1 involves traversal of all the coalitions in Ai for each player i. Therefore, running Algorithm1 takes O (|A|\*maxi ∈ [A] |Ai|) which is in polynomial time.

Comparing two coalitions takes polynomial time. And hence finding max (Pi) takes polynomial time. Checking if Γ contains coalition containing i also takes polynomial computing time. These two are nested inside a for loop. Therefore, the entire algorithm2 runs in polynomial time. Running the algorithm2 for all the profiles in Ext adds one more for loop on the outside which again falls under P. Hence algorithm6 + algorithm7 belongs to P.

**Results**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stability Concept** | **Possible**  **Verification** | **Necessary**  **Verification** | **Possible Existence** | **Necessary**  **Existence** |
| Perfect Coalition Structure | P | P | P | P |
| Individually Rational Coalition Structure | P | P | P | P |