1) Two Scenarios:

In this scenario, there are some community masters having *request rates* (R_{master}). Web services need to join a community to be able to get a task from a master web service. Each web service comes with different quality of service parameters and a throughput (T_{ws}). Throughput is the average rate of tasks a web service can perform. Therefore, the master web service is providing tasks with R_{ws} and web services perform tasks with some QoS parameters and a rate of T_{ws}.

In this setting, we define the value of the coalition as follows:

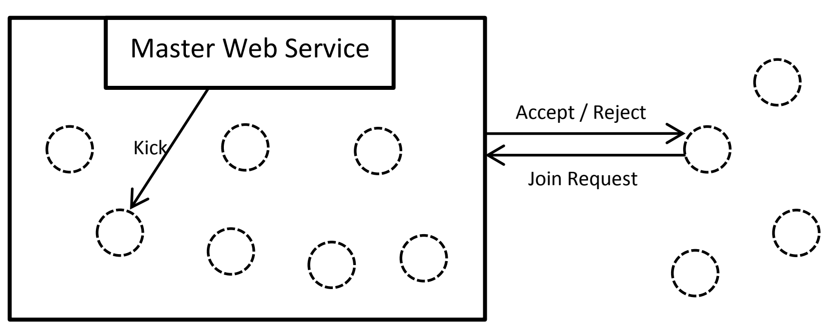
$$output(C) = \sum_{ws} T_{ws} \times\ QoS_{ws}$$

$$v(C) = \begin{cases} output(C), & \sum_{ws} T_{ws} \leq R_{master} \\ output(C) \times \dfrac{T_{ws}}{R_{master}}, & \sum_{ws} T_{ws} > R_{master} \end{cases}$$

The output of a coalition of web services if the amount and quality of task they can perform. If the amount if more than masters input task rate, it means the web services inside the community are capable of doing more job than exists. Therefore the valuation will be the amount of work they can perform considering the average QoS metrics they have. In the case where the limited tasks are distributed among web services uniformly, the value of the coalition would be the proportion of the average QoS times their throughput to rate of available requests.

1-a) Scenario 1a:

In the first scenario, we only consider one coalition or only consider the system from point of view of one master web service and a collection of web services. The master web service is the one which decides which members can join or should leave and also distributes the income or tasks among its community members.



The master web service decides on membership based on throughput and QoS of the requested web service. It will check the core membership of the coalition of all community members and the new web service. Our algorithm will use the function in formula 1 to distribute gain of *v(C)* between members and then will check if the distribution vector is member of Core or not.

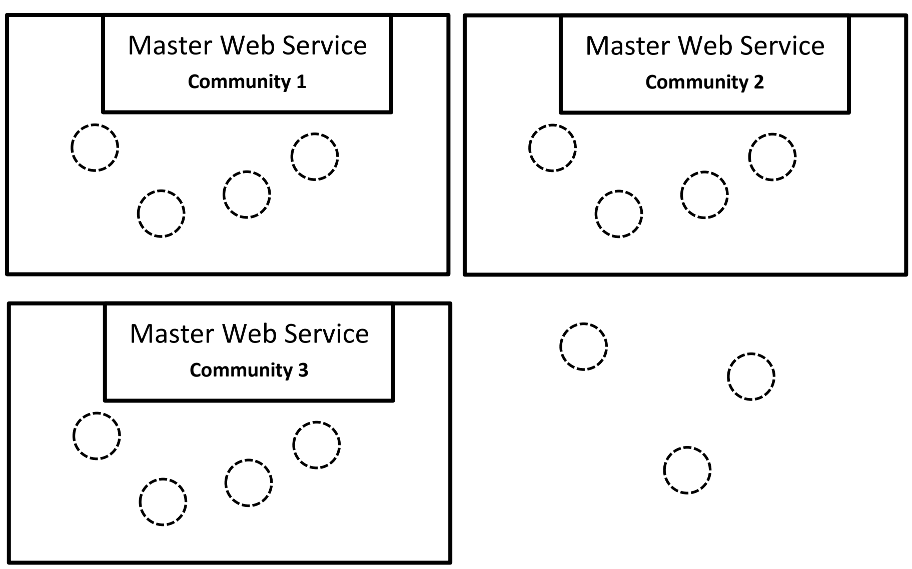
$$\varphi(i) = \sum_{i \in C}^{\infty} \left(\frac{(n - |S|!\,(|S| - 1)!}{n!} \right) (v(C) - \ v(C - \{i\}))$$

The payoff for each web service i, is calculated based on their contribution $v(C \cup \{i\}) - v(C)$ and then averaging over the possible different permutations in which the coalition can be formed (shapely value). According to [On the Complexity of the Core over Coalition Structures] if there the coalition core is non-empty, the vector of distribution [ϕ(1), … (ϕ(n)] should be member of core. And In our algorithm we just check the core member ship of this vector.

[R] Gianluigi Greco, Enrico Malizia, Luigi Palopoli, Francesco Scarcello: On the Complexity of the Core over Coalition Structures. IJCAI 2011: 216-221

1-b) Scenario 1b:

In this scenario we consider multiple communities owned by multiple master web services which provide independent request pools. Master web services form coalitions with web services. The set of web services is partitioned into non-empty disjoint coalitions namely coalition structures. There are two fundamental problems we need to address in this class of problems: 1) What coalitions will form, and 2) how will the members of these coalitions distribute their total worth.



The web services with low throughput will most likely join communities with lower request rates. Since the payment is proportional to their contribution the web services with small contribution will get payed much less in communities having web services with high throughput. On the other hand, web services with high throughput will not contribute well to communities with low amount of requests (see coalition valuation function, equation 1). So a coalition structure with non-empty core forms having stronger web services.

Another concept in coalition structure literature is maximizing social welfare. For any coalition structure π, let $v_{cs}(\pi)$ denote the total worth $\sum_{C\in\pi} \mathrm{v}(\mathrm{C})$ which we call it social welware. We want to find the maximum social welfare over all the possible coalition structures π. The algorithms for social welfare in coalition structure settings are more intractable than normal coalition structure core problems. The problem is they try to maximize social benefit while contradicts with interests of selfish self-interest web services. Since the order of computation of these algorithms are much intractable than general core problems, we can compare these solutions from individual web service point of view and see how much difference it makes.

[R] Kóczy, László Á. and Lauwers, Luc, (2002), ***The Coalition Structure Core is Accessible,*** Game Theory and Information, EconWPA, http://EconPapers.repec.org/RePEc:wpa:wuwpga:0110001.

[R] Rahwan, Talal, Michalak , Tomasz and Jennings, Nicholas (2011) ***Minimum Search to Establish Worst-Case Guarantees in Coalition Structure Generation.*** At The Twenty Second International Joint Conference on Artificial Intelligence (IJCAI), Barcelona, Spain, , 338-343.

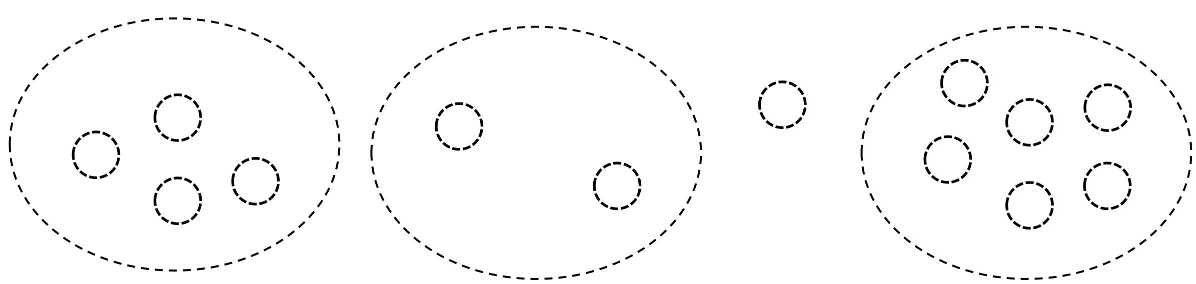
[R] Rahwan, Talal, Michalak , Tomasz, Elkind, Edith, Faliszewski, Piotr, Sroka, Jacek, Wooldridge, Michael and Jennings, Nicholas (2011) ***Constrained Coalition Formation.*** At The Twenty Fifth Conference on Artificial Intelligence (AAAI), San Fransisco, USA, , 719-725.

Also there is concept of bargaining set with coalition structure, the agents do not try to change the CS, but only obtain a better payoff. If they find out they are better of kicking a web service and gaining more they can report an objection and ask the other web service to do some payments. The agent that is targeted by the threat may try to show that it deserves his gain. To do so, its goal is to show that, if the threat was implemented, there is another deviation that would ensure that she can still obtain its gain and that no agent (except maybe the again which claims the objection) would be worse off. In that case, we say that the objection is ineffective.

[R] Robert J. Aumann and M. Maschler. ***The bargaining set for cooperative games.*** Advances in Game Theory (Annals of mathematics study), (52):217–237, 1964.

[R] Bezalel Peleg and Peter Sudhölter. ***Introduction to the theory of cooperative cooperative games.*** Springer, 2nd edition, 2007.

2) Scenario 2:



In this scenario, we do not consider pre-defined communities and master web services. Here web services are interested in forming coalitions to perform better and share benefits. We do not consider concept of masters providing requests for others, each web service comes with its own request load or here we call it market share. Therefore coalitions will form if web services working together can perform and gain more than working alone individually. The worth or value of coalition of web services is different in this setting:

$$v(C) = \sum_{ws} \left\{ \sum_{m=metrics} \left\{ \sum_{ws} QoS_{ws}^{m} \times CollabrativeCoefficient \right\} \times w_{metric} \right\} \times MarketShare_{ws}$$

Collabrative coefficient for each QoS metric determines how web services providing that QoS metric individually will be effected in a coalition environment. If it is bigger than 1, it shows they increase performance by working together otherwise it depicts the performance degrades as they work together. These values need to be extracted form real world examples or we can just analyses what numbers they need in order to be able to collaborate efficiently.

Since there are no master web services, we assume the collaborative networks agree on decisions on whether to accept a new member and how they distribute the gain. Like scenario 1-b both core coalition structure analysis and social welfare analyses are applicable in this scenario.

3) Core Analysis and Solutions

a) Background

Let N be a set of players. A coalition game with transferable utilities (a TU game) on N is a function that associates with each subset S of N (a coalition, if non empty), a real number v(S), the worth of S. Additionally, it is required that v assign zero to the empty set. If a coalition S forms, then it can divide its worth, v(S), in any possible way among its members. That is, S can achieve every payoff vector $x \in R^S$ which is feasible, that is, which satisfies

$$\sum_{i \in S} x_i \leq v(S)$$

Definition 3.1) A characteristic function game G is given by a pair (N,v), where N = {1,…n} is a finite, non-empty set of agents and $v : 2^N \rightarrow R$ is a characteristic function, which maps each coalition S ⊆ N to a real number v(S).

As mentioned, the number v(S) is usually referred to as the value or worth of the coalition S. Payoff vector should have two conditions. The first condition is *feasibility* as mentioned above and also efficiency. A payoff vector x is *efficient* if all the payoff obtained by a coalition is distributed amongst coalition members, i.e., $\sum_{i\in S} x_i \leq v(S^j) \;\; for\ every\ j \in \{1, \dots, k\}$.

b) Core

The core of an n-person game is the set of feasible outcomes that cannot be improved upon by any coalition of players. Formally, for a coalition N, the core of v is defined as the set C of all feasible payoff functions x(S), that:

$$a(S) \geq v(S), \qquad for\ all\ S \subseteq \mathrm{N}$$

If this does not hold for any S ⊆ N, then agents in S can do better by abandoning the coalition and forming a coalition of their own.

Definition 3.2) The Core of a game (N,v) denoted by C(N,v) is defined by

$$C(N, v) = \{x \ \in X(N, v)|\ x(S) \ \geq v(S), for\ all\ S \subseteq \mathrm{N}\}$$

Let x be a possible feasible payoff distribution. Then $x \ \in C(N, v)$ if and only if no coalition can improve upon x. Thus, each member of the core is a highly stable payoff distribution.

Convex Games:

A game with valuation function v(S) is convex if:

3-1) $$v(S) + v(T) \leq v(S \ \cup T) + v(S \ \cap T),\ \ for\ all\ i \in N\ and\ S \subseteq \mathrm{T} \subseteq N - \{i\}$$

In [Cores of Convex Games, Shapley] it has been proven convex games have non empty core. In order to come up with an algorithm be able to evaluate if a game is valuation function is convex or not, we will use the below equation and will prove its analogous to convexity condition:

3-2) $$v(S \ \cup \{i\}) - v(S) \ \leq v(T \ \cup \{i\}) - v(T)$$

Proof:

$$v(S \ \cup \{i\}) - v(S) \ \leq v(T \ \cup \{i\}) - v(T)$$

$$\rightarrow v(S \ \cup \{i\}) + \ v(T) \leq v(T \ \cup \{i\}) + \ v(S)$$

We know S ⊆ T, therefor:

$$(S \ \cup \{i\}) \cup T = T \cup S$$

And we have T ⊆ $N - \{i\}$, therefor:

$$(S \ \cup \{i\}) \cap T = S$$

With a variable change we have:

$$v(S \cup \{i\}) + v(T) \leq v(T \cup \{i\}) + v(S)$$

$$\rightarrow v(S \cup \{i\}) + v(T) \leq v\big((S \cup \{i\}) \cup T\big) + v\big((S \cup \{i\}) \cap T\big)$$

$$\rightarrow v(S') + v(T) \leq v(S' \cup T) + v(S' \cap T)$$

This is the convexity equation (3-1). This expresses a sort of increasing marginal utility for coalition membership which means a new member {i} has to contribute more when joining bigger groups to make keep group stable. In simulation evaluating this needs an algorithm of $O(2^N)$ worst case, where N is the number of coalition members, since we need to evaluate the equation (3-2) for all subsets of a coalition. Although usually the average order in simulation when a coalition has no core in partice, is O(N) since the equation (3-2) fails in sets {N-1} and {i} for $i \in N$.

4) Simulations

In this section, I will list the expirements I want to perform in this project. The source codes will be here
https://github.com/Marooned202/Multiagent-Simulation
https://github.com/Marooned202/wscommunity
after I finish the first phase I will commit the updated codes. I currently have developed the function to recognize if a set of coalition is forming a stable coalition (core) or not and there is a subset which everyone can gain more by deviating. This function gets a set along with a valuation function which can calculate the value of coalition base on each possible subset.
These are the main scenarios I will experiment:

1) In scenario 1-a, I will perform these experiments and compare them:
    a. We have an initial setup then we only accept or reject invitation requests from web services.
    b. We have an initial setup and we are allowed kick weak web services if kicking them will lead to a stable coalition (core) with new web service.
    c. Implement the [A Game Theoretic Approach for Analyzing the Efficiency of Web Services in Collaborative Networks, khosr.] and compare the results.
    d. Implement an stochastic random model
2) For scenario 1-b, I will compare these models:
    a. We have an initial setup then we only accept or reject invitation requests from web services.
    b. We have an initial setup and we are allowed kick weak web services if kicking them will lead to a stable coalition (core) with new web service.
    c. Finding a coalition structures which maximizes social welfare and ignoring individual
    d. Trying to keep communities (coalition structures) unchanged but keep coalitions stable by side payments using bargaining set methods.

3) In scenario 2, these experiments can be performed:
    a. We have an initial setup and we are allowed kick weak web services if kicking them will lead to a stable coalition (core) with new web service.
    b. Finding a coalition structures which maximizes social welfare and ignoring individual
    c. Trying to keep communities (coalition structures) unchanged but keep coalitions stable by side payments using bargaining set methods.

5) Challenges and Improvements

1) Being able to reduce and remodel our problem specifically to find out a different algorithm with more intractable computation complexity for determining if core of any set of web services in empty. These methods do not come of with core set, they analyze the conditions in which they can prove the core is non-empty. Therefore implying some constraints and conditions on valuation function for a specific problem we can make sure the set is stable and there can be no deviation without calculating the gain distribution among agents. This has been performed in most theoric and game theory literature. They usually perform these analyses on an abstract and simple and non-real world model and map the problem to graphs or famous algorithmic problems.

[R] Zhixin Liu, ***Complexity of core allocation for the bin packing game***, Operations Research Letters, Volume 37, Issue 4, July 2009, Pages 225-229,

[R] Aadithya, Karthik V., Ravindran, Balaraman, Michalak, Tomasz P. and Jennings, Nicholas R. (2010) ***Efficient Computation of the Shapley Value for Centrality in Networks.*** In, Proc 6th Int Workshop on Internet and Network Economics (WINE 2010), Stanford, 13 - 18 Dec 2010. , 1-13.

[R] Coralio Ballester, ***NP-completeness in hedonic games, Games and Economic Behavior***, Volume 49, Issue 1, October 2004

[R] Anna Bogomolnaia, Matthew O. Jackson, ***The Stability of Hedonic Coalition Structures***, Games and Economic Behavior, Volume 38, Issue 2, February 2002

[R] Milan Maros, ***Sufficient conditions for the solution existence in general coalition games,*** KYBERNETIKA- VOLUME 21, NUMBER4

2) Experiment different valuation functions for collaborative web services.