MDP BASED OPTIMAL POLICY FOR COLLABORATIVE PROCESSING USING MOBILE CLOUD COMPUTING

MONA NASSERI (UT), *ROBERT GREEN* (BGSU), AND MANSOOR ALAM (UT)

UNIVERSITY OF TOLEDO (UT)

BOWLING GREEN STATE UNIVERSITY (BGSU)

PROBLEM STATEMENT

Question:

- How can mobile phones collaborate with each other in order to complete a particular task in a more efficient manner?

Answer:

- Through a combination of **Mobile Cloud Computing**, **Collaborative Networking**, and **Markov Decision Processes** and look-up tables (of course)!

MOBILE CLOUD COMPUTING

Definition:

- A combination of **cloud computing** and **mobile environments**

✓ Useful for **off-loading** and **sharing** the various burdens related to complex computation and/or data storage.

✓ **Offloading** (or Cyber foraging) enables the mobile devices to offload tasks by **leveraging unused sources** on larger computers

COLLABORATIVE NETWORKING

Definition:

- A collaborative network refers to an **ad-hoc network** system that is formed by users in **close proximity** to one another

✓ **Pooling** their resources

✓ **Reducing overall load** on a single device by using the other devices as mobile data relays.

MARKOV DECISION PROCESS

- MDP is a promising solution to combat calculation complexities as a mathematical framework
- Used to create decision tables, including outcomes which are partly random and partially dependent on user decisions
- MDP has a decision agent which checks the current state, **s**, repeatedly, take the decision to do action ***a*** with probability ***p*** which leads to the transition to state **s′** including a reward, ***r***

MARKOV DECISION PROCESS

MDP Parameters

- **S - State Space:** All possible states of the system, which are known to the decision-maker.
- **A -** All possible actions that can be taken by the decision-maker
- **R - Reward:** The reward for taking action *a* in a state s**.**
- **P - Transition Probability:** The probability that an action *a* taken in state s at time t will result in a transition to state s′ in time t + 1.

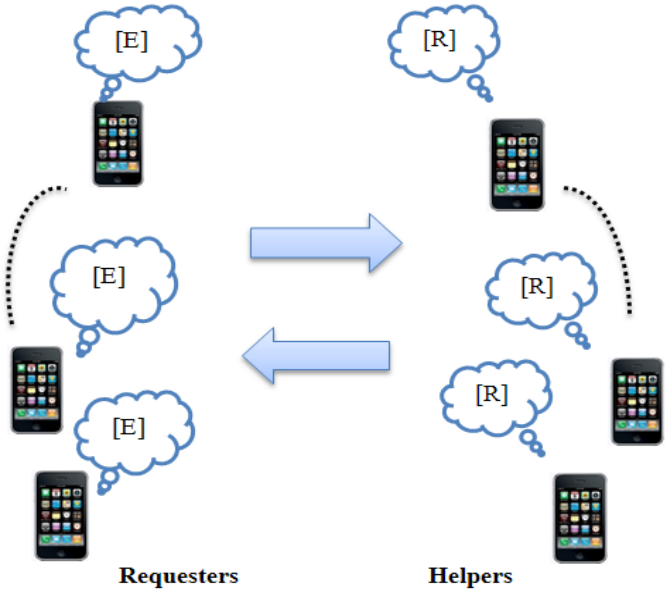
PROPOSED METHOD

Collaborative downloading

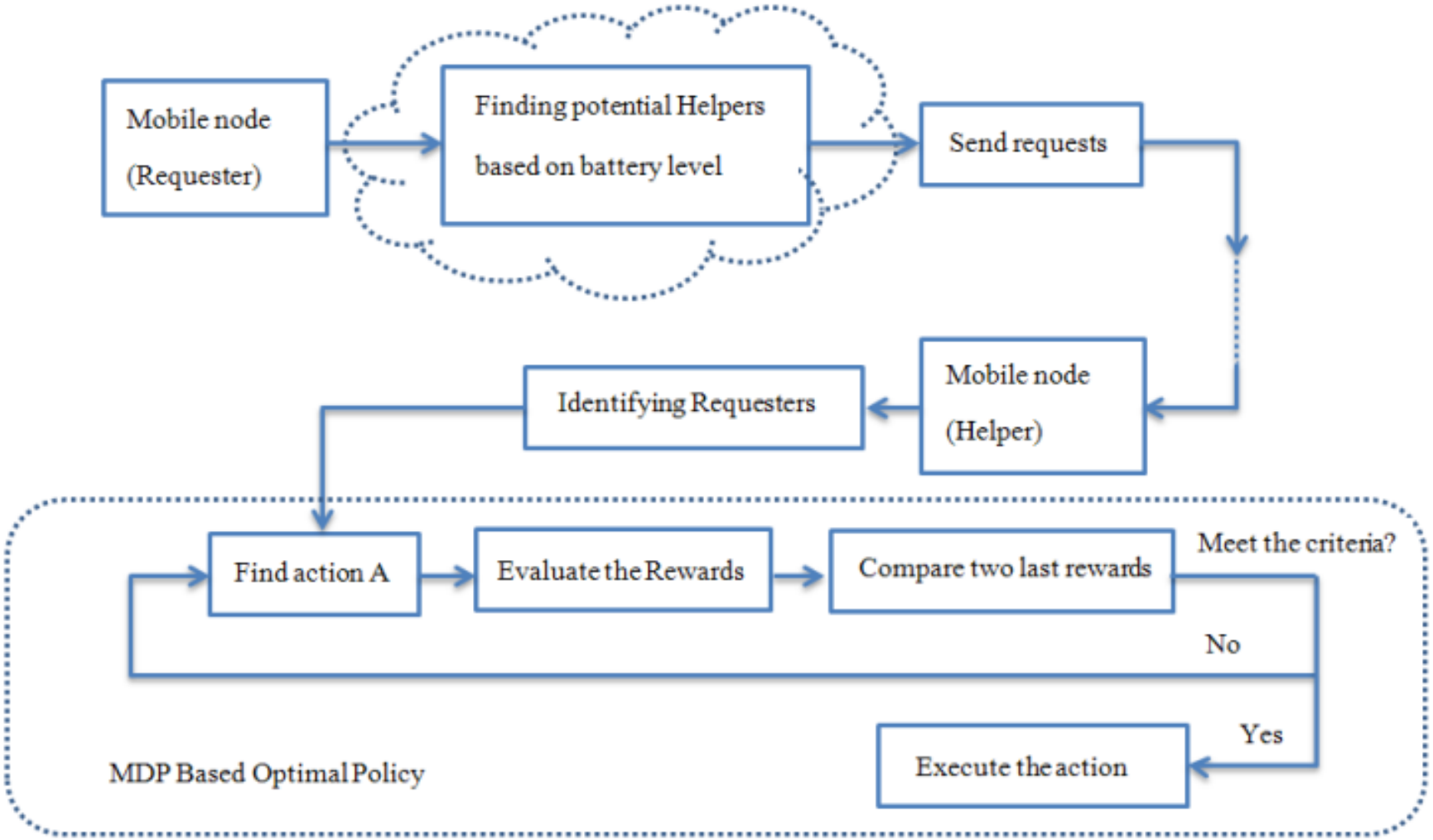
- There are ***n* phones.** Some ask other mobile devices to help in the downloading process.

Helpers can

- **Accept** the request and collaborate
- **Reject** the request to download the file
- **Relay** in order to send a file to a destination.



PROPOSED METHOD



REQUESTER SIDE POLICY

- The requester's decision is established on a **threshold policy** that is based on an individual phone's determination of how conservative it wants to be in saving its charge for future communications.

- Each phone determines its $\mathbf{E_{th}}$ **(energy-threshold)** and $\mathbf{E_i}$ **(current energy level)** and sends it to service provider.

- The requesting phones use the **server's look up tables** in order to choose which helper should send a request.

$$E = \begin{bmatrix} E_{i1} & E_{th1} \\ \vdots & \vdots \\ E_{in} & E_{thn} \end{bmatrix}$$

REQUESTER SIDE POLICY

If $\mathbf{E_i - E_{th} > e_o + e_d + e_f}$**, then** its identification term will be saved at $\mathbf{E_{sel}}$ matrix according to their conditions from excellent to fair

- $\mathbf{e_o}$ **:** Energy overhead for establishing collaboration
- $\mathbf{e_d}$ **:** Download energy cost
- $\mathbf{e_f}$ **:** Energy for helper to forward download

$$E_{sel} = \begin{bmatrix} k = excellent \\ \vdots \\ m = fair \end{bmatrix}$$

REQUESTER SIDE POLICY

- The matrix, ***E***, is saved at the server and is updated each *T* minutes.
- $\boldsymbol{E_{sel}}$ will be sent to the requester in order to aid in **choosing the helper phone**.
- Messages are only sent to those **potential helpers** identified by the requestor.

HELPER SIDE POLICY

- The helper phone must decide to **accept** or **reject** the request that is presented by a requester.

- If the number of requests increases, the helper can **choose one request** according to calculated rewards.

- In an environment that includes **several requests**, a helper can **accept one** request and **reject others** or **reject all** of them based on the results of the MDP.

HELPER SIDE POLICY

MDP Parameters:

- **A**=$\{a_{i,j}\}\epsilon$ {0, 1}
- **s** ϵ **S**{P, N, T}
  - **P** = {1, 2, 3, …..p_{max}} in mw
  - **N** ={1, 2, 3} number of bars or received signal code power (RSCP) level; and
  - **T** = Time since last recharge

HELPER SIDE POLICY

Reward Components

Power Reward

$$f_p(s,a) = \frac{1}{1+\exp(p_a)}$$

Delay Reward

$$f_d(s,a) = \frac{1}{1+\exp(d_a)}$$

Transition Cost Function

$$h(s,a) = \begin{cases} H_{i,j} & i \neq j \\ 0 & i = j \end{cases}$$

HELPER SIDE POLICY

Reward Function

$$\mathbf{f(s,}\boldsymbol{a}\mathbf{) = w_p \times f_p(s,}\boldsymbol{a}\mathbf{) + w_d \times f_d(s,}\boldsymbol{a}\mathbf{)}$$

$$\sum_m w_m = 1$$

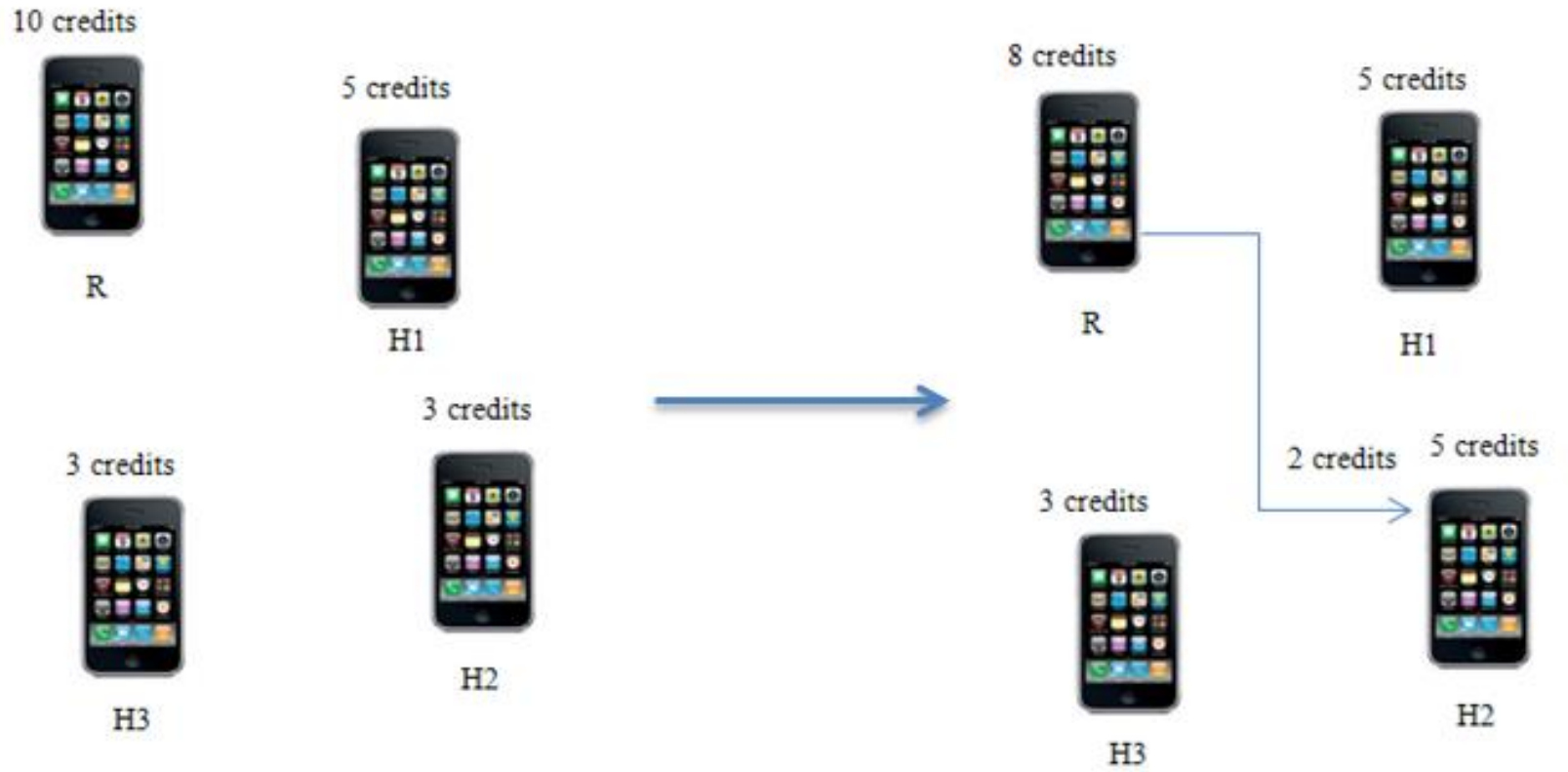
$$\mathbf{r(s,}\boldsymbol{a}\mathbf{) = f(s,}\boldsymbol{a}\mathbf{) - h(s,}\boldsymbol{a}\mathbf{)}$$

CREDITS

$$C(r) = \ln(r) + 1$$

- Should be scaled in **credit domain ($credit_{min}$, $credit_{max}$).**
- 1 is added to **show each activity includes credit.**

CREDIT EXCHANGE



RESULTS

Initial Results

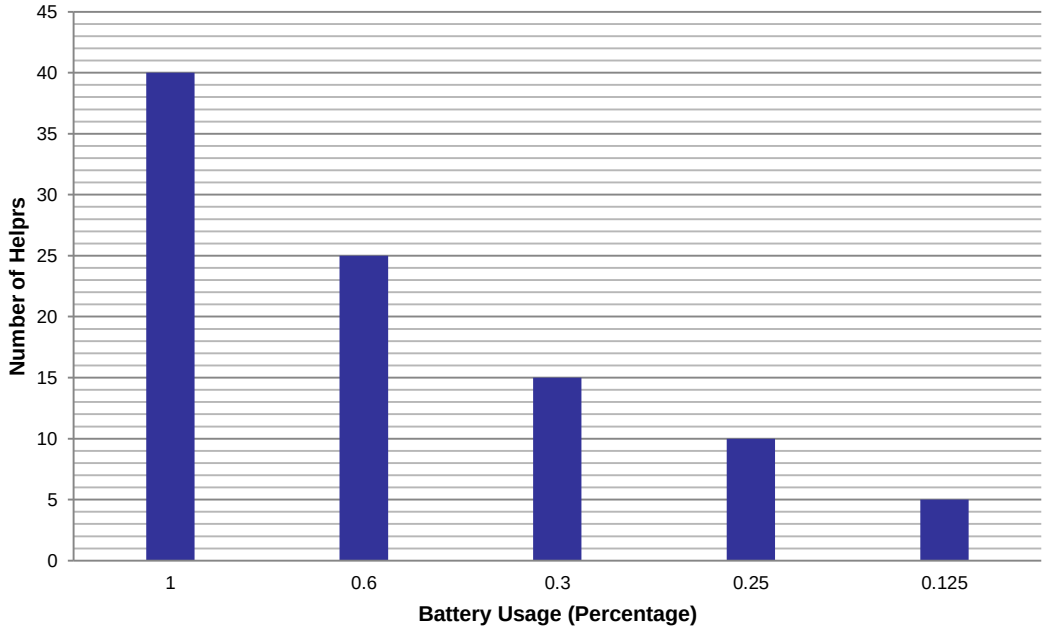
- Impact of Helper Requests
- Impact of Power Reward
- Impact of Delay Reward

Simulation Results

- Simulation Network
- Rewards under Varying Power Consumption
- Credits Received under Varying Power Consumption

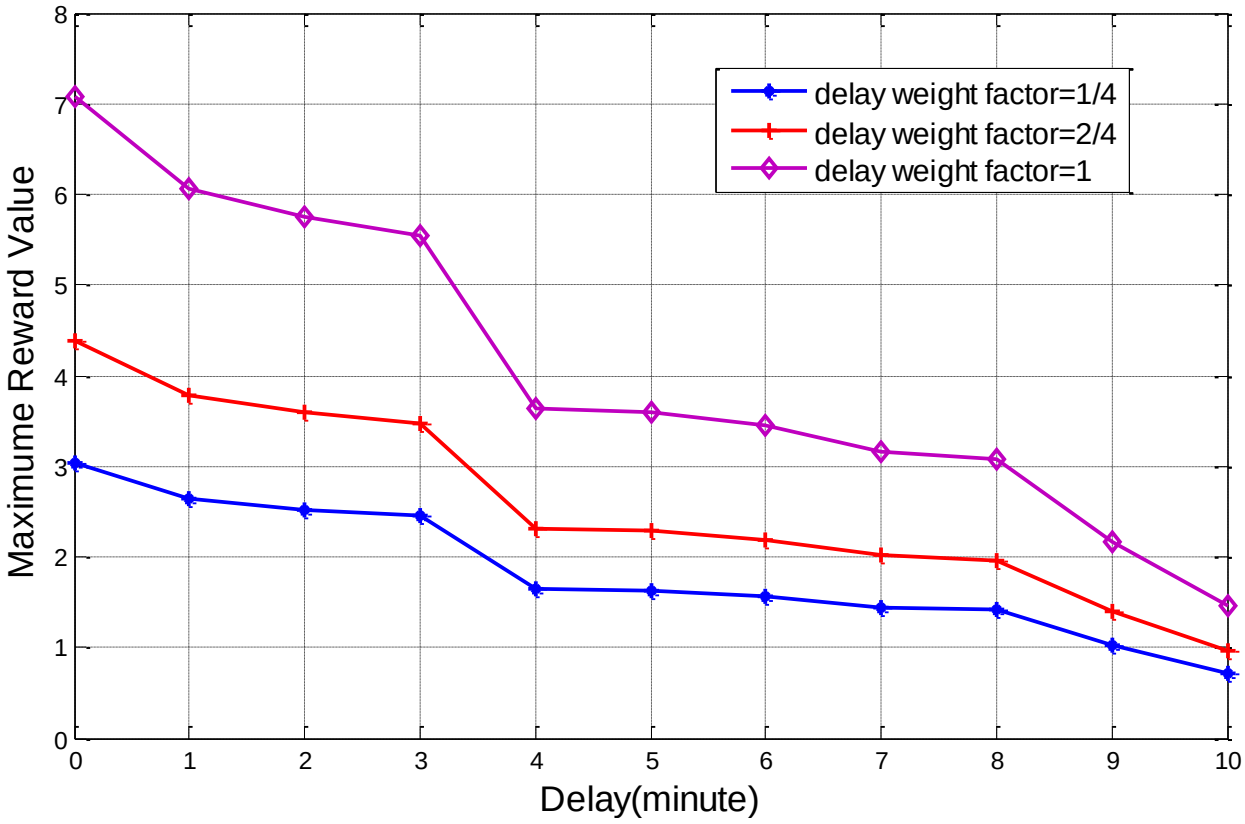
INITIAL RESULTS

A message with content of “Download Request” is sent to different Iphone 4s using a 3G network.



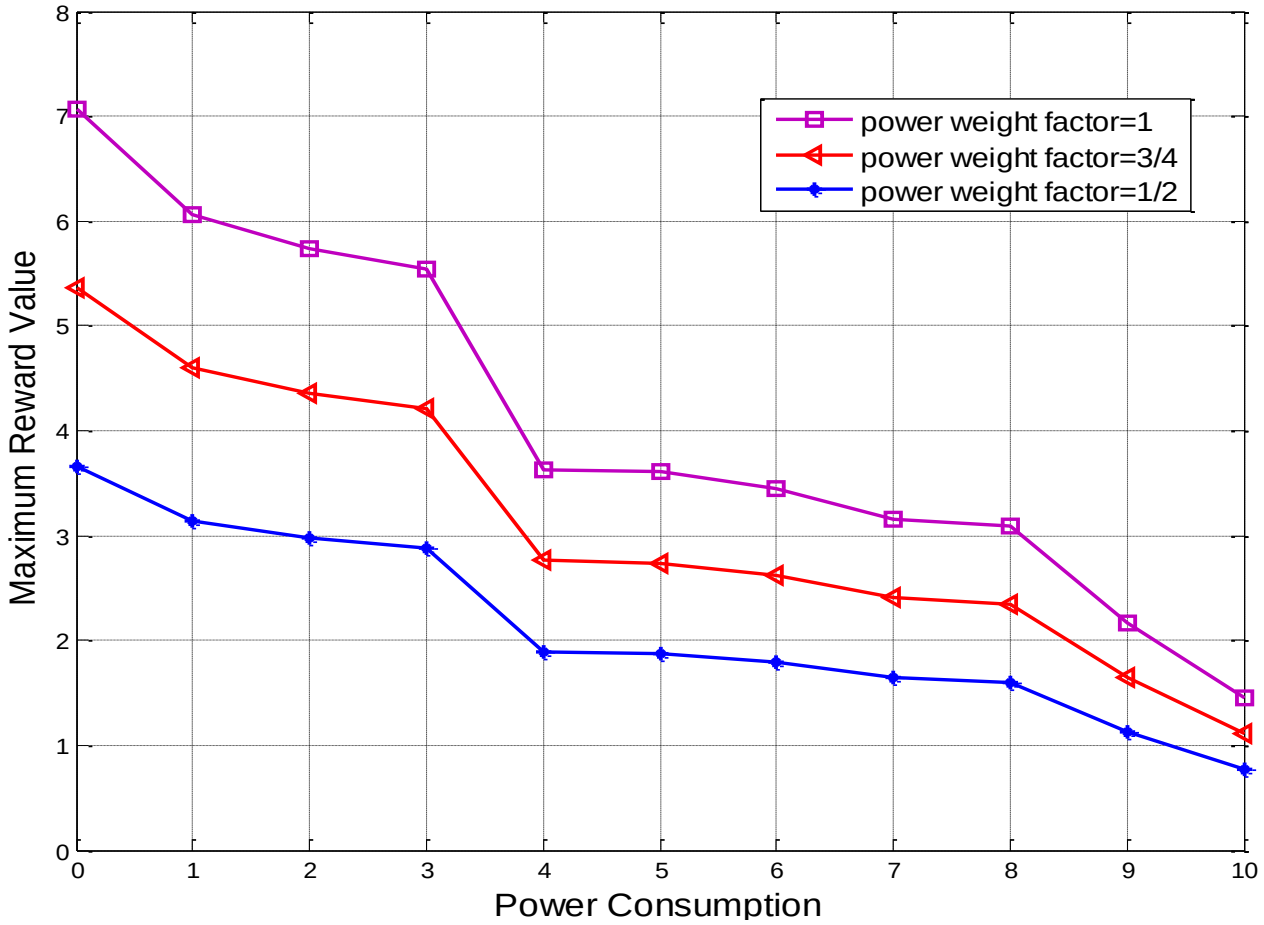
INITIAL RESULTS

Fixed Power Consumption



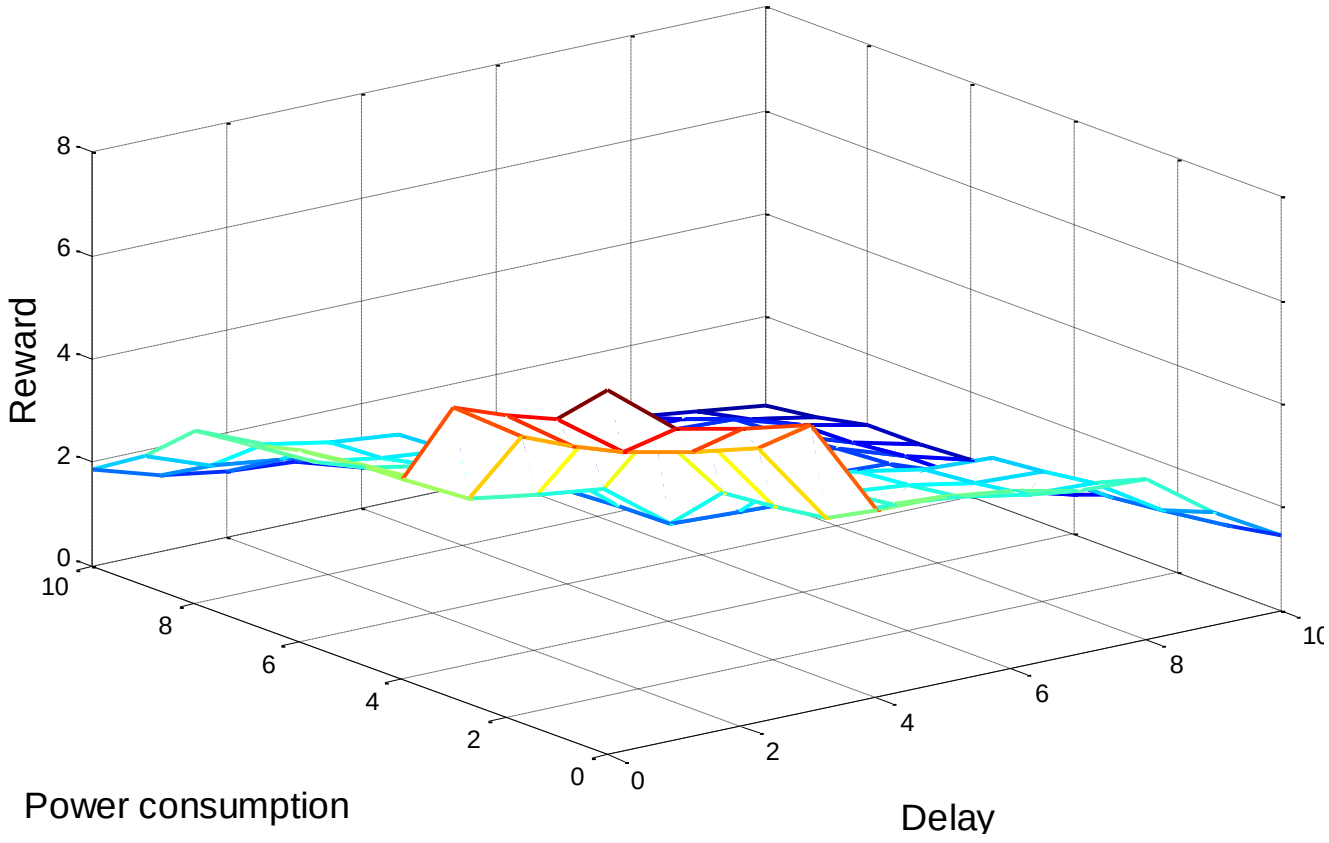
INITIAL RESULTS

Fixed Delay



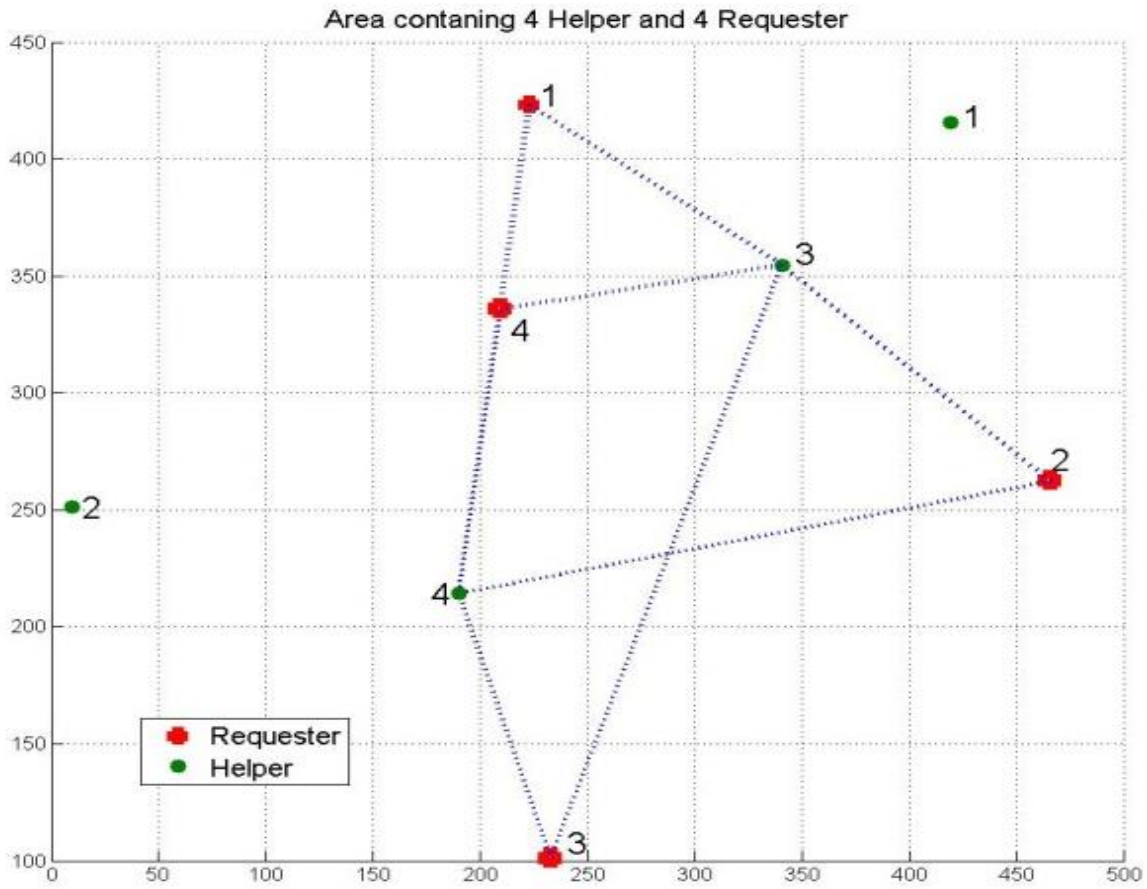
INITIAL RESULTS

Relation between power, delay, and reward



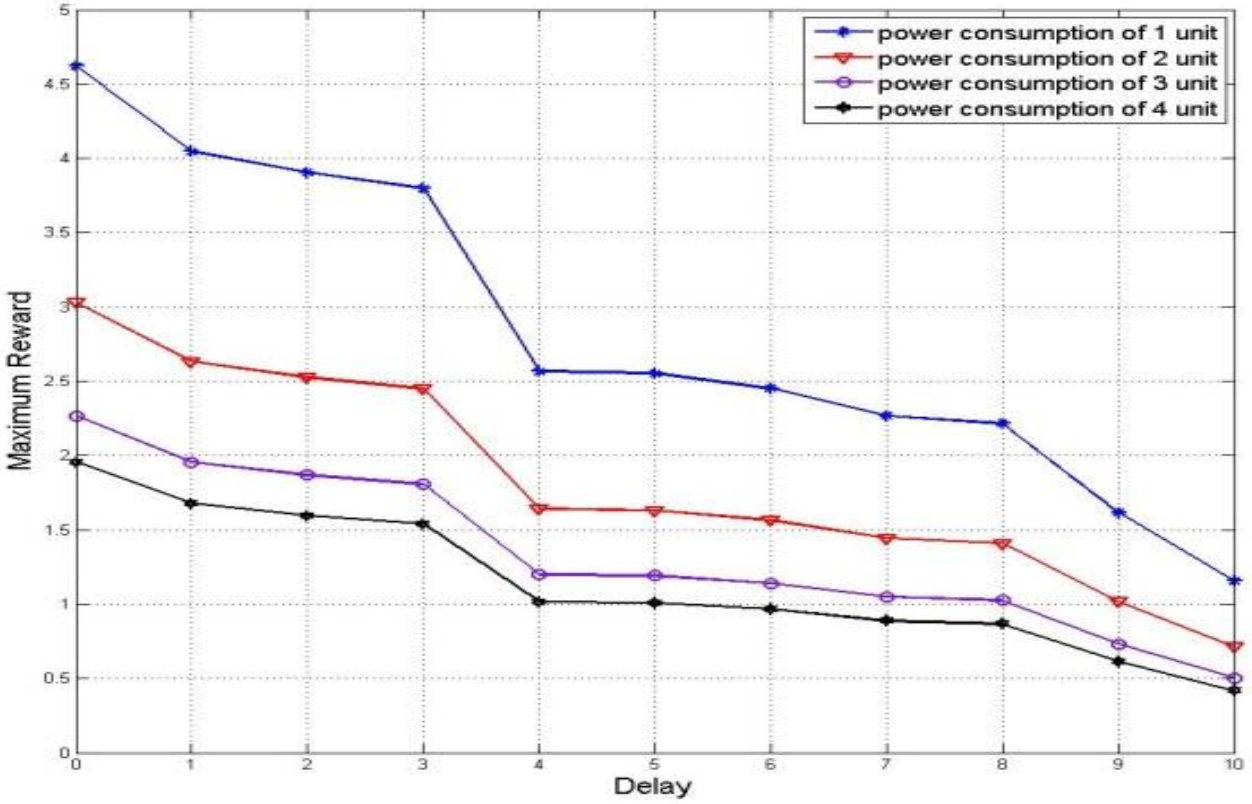
RESULTS

Simulated Network



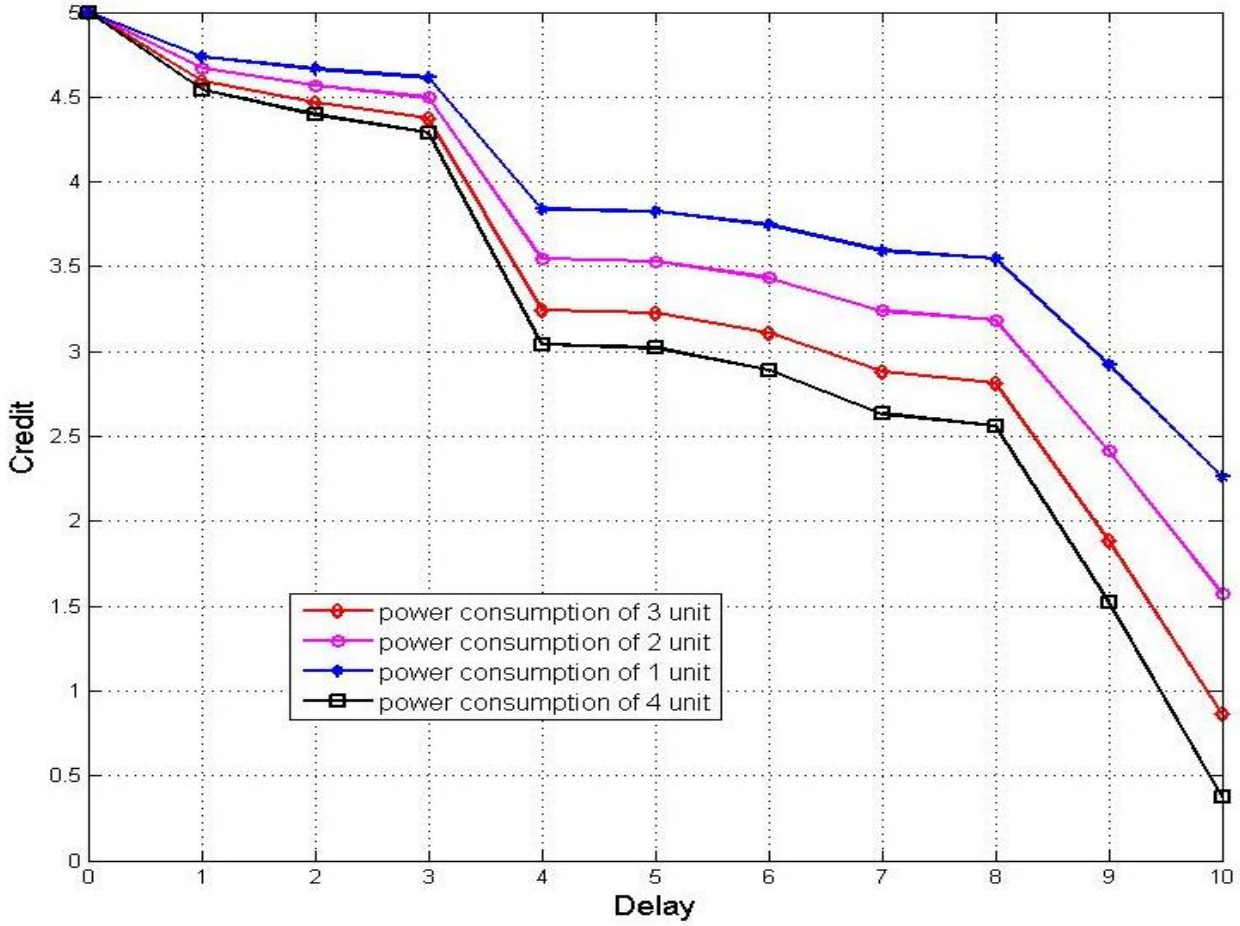
RESULTS

Maximum Reward Comparison



RESULTS

Credit Evaluation



SUMMARY AND CONCLUSION

- **Optimal policies** for mobile cloud computing on both the **requester** and **helper** sides are presented
- The policy on requester side is based on differences **of energy threshold** and **battery level** of the helper mobile device.
- The policy on helper side is based on **MDP** and **maximum calculated reward** through iteration algorithm.
- Simulation shows less delay at responding to a request and less power consumption, resulting in higher amount of rewards.
- Potential future work may include applying **SMDP instead of MDP** in order to achieve more realistic results, evaluating larger networks, and other applications

Thank you