continues until agents have converged, scheduling themselves in 1) time and 2) reliability.

In our model, each agent has a vector of desired activities that it would like to perform. The activities are defined similar to [3], however with the addition of a *tier* field that indicates a quality of service preference:

$$A_{desired,i,k} = \{lb, ub, P(t), tier\} \quad (1)$$

where $A_{desired,i,k}$ represents the k^{th} desired activity of agent i, lb is a lower bound of the time i would like A to start, ub is an upper bound on the time, $P(t)$ is the activity's power profile as a function of time, and $tier$ represents a quality of service associated with the activity. The $tier$ variable determines the utility function that an agent will use to bid for an activity's reliability.

To schedule activities, we leverage work in [4] and use a risk-based tatonnement algorithm. In traditional tatonnement the good is normally a physical good such as corn, oil, or electricity. In our formulation, the good is reliability. This allows us to account for the stochastic nature of solar generation; no load can be served with certainty. We define the reliability of A as the probability that the network is able to serve the activity over a specified time horizon, $T = \{t_1, ..., t_N\}$. For example, if $P(A_{actual,1}) = 0.75$ and the scheduling horizon is one day, then the grid can serve Activity 1 on 75% of days. The form of the bid is the 3-tuple, $bid_i = \{startTime, price, P(t)\}$ where $bid.startTime$ refers to when agent i would like its activity to start, $bid.price$ refers to the price the agent is willing to pay for reliability, and $bid.P(t)$ is the activity's power profile as a function of time.

Each activity has associated with it a local utility function that relates to its quality of service. Similar to [4], the utility functions considered are quadratic functions linear in price:

$$U_i(t, r) = a_i r^2 + b_i r - p(t, r)r \quad (2)$$

In this equation, constants a_i and b_i represent an agent's desire for reliability (i.e. the user's tier of service), r represents the reliability in decimal form, and $p(t, r)$ is the price for reliability. In our implementation in India, the tier and corresponding utility function will be assigned by the grid operator to the user, who pays a certain premium for a higher tier.

After nodes submit bids, it must be determined whether they are feasible. That is, given a set of bids, $\mathbf{B}$, it must be determined whether $P(A_{actual,i}) = P(A_{demanded,i})$ for all bids. For example, if $P(A_{actual,1}) = 0.75$ and $P(A_{demanded,1}) = 0.90$, then this bid is not feasible and Activity 1's price must be raised for the next bidding round. Bid feasibility is determined by a leader node where we use two innovations. First, the leader node imposes the notion of fairness through a probabilistic ordering on activities where nodes who bid higher are more likely to receive power in the case of a scheduling conflict. Second, we use a simple Monte Carlo method to determine if bids are feasible under multiple solar generation scenarios. Price updates continue until convergence, similar to traditional tatonnement. Once converged, agents know when and with what probability their activity will be served.

3. MARKET FOR RELIABILITY TEST CASE

A test case was performed to illustrate the algorithm. The test case consisted of twelve activities of two types: lighting and cell charing (typical activities for Indian villages). Each type of activity had the following time constraints and power characteristics: Charging: $\{lb = 0, ub = 12, duration = 3, power = 5\}$ and Lighting: $\{lb = 6, ub = 12, duration = 2, power = 8\}$. The scheduling horizon was a twelve hour day. Cell charging could take place over the entire day while lighting could only take place at night. Activities were divided into two tiers, where Tier 1's activities were served with the greatest probability. Figure 2 shows the output schedule. Tier 1 activities receive higher reliability than Tier 2 activities. Although they are unconstrained in time, agents uniformly decided to schedule cell charging activities during the day because lighting activities are constrained to only happen in the evening and must compete for energy.

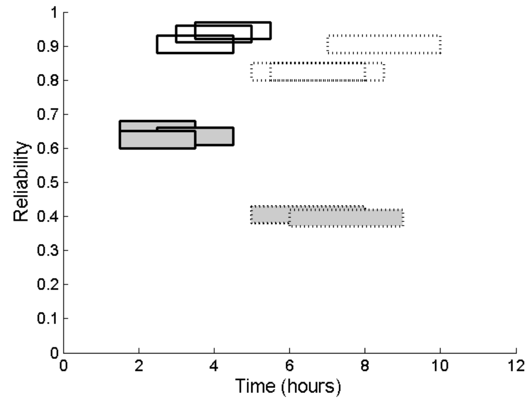

Figure 2: Results for a 12-activity test case. Tier 1 activities have no fill and Tier 2 activities are filled in gray. Cell charging has solid outlines and lighting activities are dashed outlines.

4. CONCLUSION

The Market for Reliability algorithm attempts to bridge the gap between stochastic programming and multi-agent market-based approaches. The algorithm takes into account users' desires for power-related activities and their preferences for reliability and outputs a feasible schedule. Currently, we are working on piloting our microgrid in a village in eastern India while implementing the M.f.R. algorithm. Future work will focus on the results of this implementation.

REFERENCES

[1] International Energy Agency. *World Energy Outlook*. International Energy Agency, Web, 2013.

[2] Andy Campanella. An analysis of the viability and competitiveness of dc microgrids in northern india. Master's thesis, MIT, 2013.

[3] Menkes van den Briel, Paul Scott, and Sylvie Thiebaux. Randomized load control: A simple distributed approach for scheduling smart appliances. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Beijing, China, August 2013.

[4] Masaki Yo, Masahiro Ono, Brian C. Williams, and Shuichi Adachi. Risk-limiting, market-based power dispatch and pricing. In *European Control Conference*, July 2013.