Fast Precise Distributed Control for Energy Demand Management

Jacob Beal
Raytheon BBN Technologies
Cambridge, MA, USA, 02138
Email: jakebeal@bbn.com

Jeffrey Berliner
Raytheon BBN Technologies
Cambridge, MA, USA, 02138
Email: berliner@bbn.com

Kevin Hunter
Silver Bay Software LLC
Dunstable, MA, US, 01827
Email: kevin.hunter@silverbaysoftware.com

***Abstract*—Fast and precise demand shaping is critical for the electrical power grid. With residential and small-business customers, a distributed approach to demand shaping is desirable for reasons of scalability and of privacy. The *ColorPower* architecture [1] provides such an approach, but the controller previously used was badly limited. We now present an improved control algorithm, ColorPower 2.0, based on stochastic constraint satisfaction, which provides major improvements in capability and performance over the prior algorithm. Analysis shows that its performance is within a small constant factor of optimal, and these results are confirmed empirically on simulated networks of 100 to 1 million devices.**

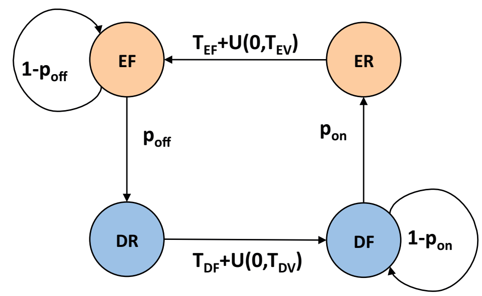


Figure 1. ColorPower devices switch states according to a modified Markov model: between **Enabled** (**E**) and **Disabled** (**D**) probabilistically and from **Refractory** (**R**) to **Flexible** (**F**) by a randomized timeout.

I. Introduction

Matching supply and demand on the electrical power grid has mainly been done on the supply side, adjusting generation to match fluctuations in consumer demand. Demand shaping adjusts consumption instead. This is important for preventing blackouts and brownouts when generation or transmission capacity is insufficient, for avoiding extreme price spikes in periods of peak demand (e.g., hot summer afternoons), and for coping with the variability of renewable sources such as solar and wind. In these and other ways, fast and precise demand shaping is key to future energy security.

A major opportunity and challenge for demand shaping is residential and small-business customers, as overall they consume a plurality of power [2], but typically only a few kilowatts per customer. A number of studies (e.g., [3], [4]) have found much flexibility in these customers' consumption, but prior approaches have been unable to take advantage of this opportunity, either because they are designed for large industrial or commercial systems (e.g., [5], [6]) or only operate well with small numbers of devices (e.g., [7], [8], [9]). The challenge is to both be acceptable to customers and scalable to millions of customers.

We have previously begun to address this with the *ColorPower* architecture for distributed demand shaping [1], [10], in which customers indicate flexibility by setting participating electrical devices to one of a few "colors." Our initial formulation has four colors ("green" devices can be shut off at any time, "yellow" only at peak power, "red" in emergencies, and "black" can never be shut off), plus a temporary customer override. A ColorPower agent for each customer aggregates device flexibility information, which is further aggregated to form a shared model of total flexibility. Individual agents then use this model in a distributed controller that adjusts aggregate demand toward the target. The prior controller had serious drawbacks, however: parameters had to be tuned for each network, sometimes there were significant overshoots, and it could not handle steep ramps.

We now present an improved control algorithm, ColorPower 2.0, based on stochastic constraint satisfaction, which has none of the problems of the prior algorithm. Analysis shows that its performance is within a small constant factor of optimal, and these results are confirmed empirically on simulated networks of 100 to 1 million devices.

II. The ColorPower Architecture

The general *ColorPower* architecture automatically matches the demand shaping requests of a utility (or other control authority) with qualitative flexibility preferences marked by customers as "colors." Local measurements are then aggregated to form a shared summary model that is used for distributed shaping of demand.

A. Device States and Transitions

Within each color, every device in the network is either **Enabled**, meaning that it can draw power freely, or **Disabled**, meaning that is has been shut off.[1] Devices must not switch too rapidly between **Enabled** and **Disabled**, as this may annoy the customer or may damage some devices,

[1]Devices with a "low power" mode can be split into two virtual devices.

978-0-7695-4851-7/12 $26.00 © 2012 IEEE
DOI 10.1109/SASO.2012.12

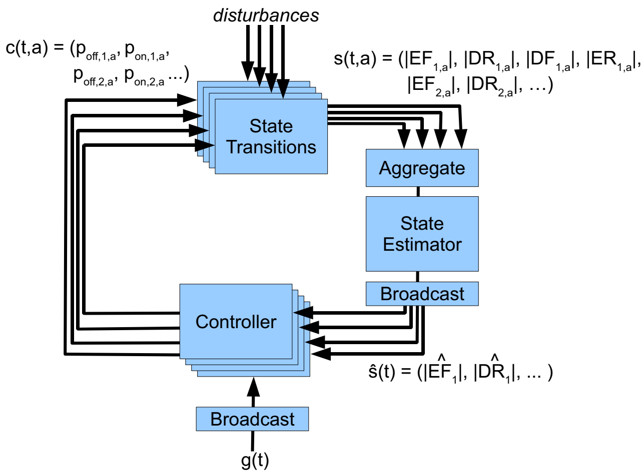


Figure 2. Block diagram of ColorPower control architecture: each agent runs a controller that regulates the state transitions of the devices under its control. Per-agent state $s(t, a)$ is aggregated to produce a global state estimate $\hat{s}(t)$, broadcast along with goal $g(t)$, to allow agents to independently set control state $c(t, a)$ to produce the desired overall shaping.

so when a device switches state it temporarily enters a **Refractory** mode where it cannot switch states. When this times out, the device is **Flexible** again and can switch states.

Each device thus evolves according to the modified Markov model shown in Figure 1. Each round, devices in state **EF** (**Enabled** and **Flexible**) randomly switch off with probability p_{off}, transitioning to state **DR** (**Disabled** and **Refractory**). Once in **DR**, a device waits for $T_{DF} + U(0, T_{DV})$ rounds before transitioning to state **DF**, where $U(0, T_{DV})$ is a uniform random variable used to desynchronize device switching. The other two distributions are complementary.

B. The ColorPower Control Problem

The control problem for an entire ColorPower system is shown by the block diagram in Figure 2: a set of n agents controlling a set of electrical devices organized into k colors, where lower numbered colors are shut off first.

The state $s(t, a)$ of an agent a at time t summarizes the power demands of the devices under its control, e.g., $|EF_{1,a}|$ is the demand power (watts) currently **Enabled** and **Flexible** for color number 1 at agent a. These values are aggregated using a distributed algorithm (e.g., a spanning tree in [1]) and fed to a state estimator to get an overall estimate $\hat{s}(t)$ of the true state $s(t)$ of demand in each state for each color, e.g., $|\hat{EF}_1|$ is the estimate of the total **Enabled** and **Flexible** demand for color 1. This estimate is then broadcast to all agents (e.g., by gossip-like diffusion in [1]), along with the goal $g(t)$ for the next total **Enabled** demand (summing across all colors). Finally, the controller at each agent a sets the transition probabilities $p_{on,i,a}$ and $p_{off,i,a}$ for each color i. We term this set of probabilities the control state $c(t, a)$.

The control problem is to set each $c(t, a)$ so the total **Enabled** demand in $s(t)$ tracks $g(t)$, subject to the constraints that: 1) devices with lower numbered colors should be shut off before devices with higher numbered colors, 2) every device in a color is equally likely to be **Disabled**, and 3) devices within a color trade off which are **Enabled** and which are **Disabled** over time, such that no device is unfairly burdened by its initial bad luck in becoming **Disabled**.

This paper focuses on the controller, assuming aggregation, estimation, and broadcast yield a model accurate within a fixed estimation error ϵ, a reasonable assumption given the wide range of deployed networks and available algorithms.

Note devices are controlled indirectly, by setting the p_{on} and p_{off} values that affect their evolution. This is unusual for a control problem and rules out most common control methods. In [1], a simple PID controller regulated p_{on} and p_{off}. With network-tuned parameters and an 5 second rounds, the prior controller gave response times on the order of 1000 seconds (200 rounds), even when orders of magnitude faster responses should have been possible. In the next section, we introduce our new constraint-based controller, which scales without tuning to networks of arbitrary size and which provides near-optimal convergence times.

III. ColorPower 2.0 Controller

To formulate an improved controller, we begin with mathematical expression of our four control constraints:

- **Goal tracking** may be expressed as:

$$g(t) = \sum_i |EF_i| + |ER_i|$$

  meaning the sum of **Enabled** demand equals the goal.
- **Color priority** may be expressed as:

$$|EF_i| + |ER_i| = \begin{cases} D_i - D_{i+1} & \text{if } D_i \leq g(t) \\ g(t) - D_{i+1} & \text{if } D_{i+1} \leq g(t) < D_i \\ 0 & \text{otherwise} \end{cases}$$

  where D_i is the demand for the ith color and above:

$$D_i = \sum_{j \geq i} |EF_j| + |ER_j| + |DF_j| + |DR_j|$$

  meaning devices are **Enabled** from the highest color down until the goal is reached.
- **Fairness** may be expressed as:

$$\forall_{a,a'} c(t, a) = c(t, a')$$

  meaning agents are identically controlled (agents may actually control differently, so long as the *expected* effect is identical—a key reason for distribution).
- **Cycling** may be expressed as:

$$(|EF_i| > 0) \cap (|DF_i| > 0) \implies (p_{on,a,i} > 0) \cap (p_{off,a,i} > 0)$$

  meaning as long as there are both **Enabled** and **Disabled Flexible** devices, some should be changing from **Enabled** to **Disabled** and vice versa. This is true only for the boundary color b (where $D_{b+1} \leq g(t) < D_b$).

A. Controller Design

It will often be impossible to satisfy all constraints, so we prioritize based on how problematic it is when they are not satisfied. As customer participation is a *sine qua non*, promises to customers are highest priority: fairness and the qualitative guarantees of colors (e.g. "only during emergencies"). Fairness can be satisfied by computing $c(t,a)$ symmetrically at every agent.[2] The qualitative priority guarantee (stating that the entirety of a color is **Enabled**) is treated as a hard requirement that no device be switched off in any color not ultimately required to satisfy the goal. Second is goal tracking (the reason for demand shaping), then soft color priority ensuring that any inversions of customer preferences caused by goal tracking are transitory. Least important is cycling, which operates slowly over time.

Because the controller acts on p_{on} and p_{off}, only **Flexible** devices can be switched on or off. The controller this has a "budget" of flexibility to spend, with each color i offering $|EF_i|$ watts of potential reduction in demand and $|DF_i|$ watts of potential increase. Flexibility accumulates as **Refractory** devices finish their timeouts and become **Flexible**, and is expended as the controller causes them to switch between **Enabled** and **Disabled** states. Our controller is thus a cascade of "flexibility budget" allocations. Constraints are considered in priority order, each allocating flexibility to satisfy itself until either all are satisfied or all flexibility is either spent or reserved against expected future needs.

1) First Allocation: Goal Tracking (constraint "g"): Goal tracking uses a proportional controller (to avoid overshoots, which **Refractory** states make problematic). Each round it tries to correct α fraction of the current tracking error:

$$C^g = \alpha \cdot (g(t) - \sum_i |\hat{EF_i}| + |\hat{ER_i}|)$$

(note the switch to estimated state). The desired downward shift Δ_i^{g-} in **Enabled** demand for the ith color is thus:

$$\Delta_i^{g-} = \begin{cases} 0 & \text{if } C^g \geq 0 \text{ or } i > b \\ |\hat{EF_i}| & \text{else if } \sum_{j \leq i} |\hat{EF_j}| \leq |C^g| \\ |C^g| - \sum_{j<i} |\hat{EF_j}| & \text{else if } \sum_{j<i} |\hat{EF_j}| < |C^g| \\ 0 & \text{otherwise} \end{cases}$$

The first case says not to reduce demand if the desired correction is positive or if this color is ineligible for reduction. The second and third cases consume flexibility from the lowest color upward toward the boundary color in order to move a total of $|C^g|$ toward the goal. The final case does nothing if the desired correction has already been satisfied.

The desired shift upward Δ_i^{g+} is the converse, consuming flexibility from the highest color downward when the goal is above the current **Enabled** demand. Moving upward, however, there is no $i > b$ constraint, since is always permissible to stop exerting control over a device.

2) Second Allocation: Color Priority (constraint "p"): This allocation ensures non-boundary colors are entirely **Enabled** or **Disabled**. This draws on unallocated flexibility:

$$|\hat{EF_i}|' = |\hat{EF_i}| - \Delta_i^{g-} \quad \text{and} \quad |\hat{DF_i}|' = |\hat{DF_i}| - \Delta_i^{g+}$$

Each round the control will try to move **Enabled** demand below the boundary and **Disabled** demand above the boundary into the boundary color, prioritizing on the most extreme colors. The desired downward shift Δ_i^{p-} in **Enabled** demand for the ith color (except b) is thus equal to:

$$\Delta_i^{p-} = \begin{cases} 0 & \text{if } i \geq b \text{or } \sum_{j \leq i} |\hat{EF_j}|' > |DF_b|' \\ |\hat{EF_i}|' & \text{else if } \sum_{j \leq i} |\hat{EF_j}|' \leq |DF_b|' \\ |DF_b|' - \sum_{j<i} |\hat{EF_j}|' & \text{else if } \sum_{j<i} |\hat{EF_j}|' < |DF_b|' \end{cases}$$

and for color b the desired upward shift is $\Delta_b^{p+} = \sum_{i \neq b} \Delta_i^{p-}$. The shift of **Disabled** demand from colors below b into color b is precisely the converse.

3) Final Allocation: Cycling (constraint "c"): Cycling moves devices fairly between **Enabled** and **Disabled** in the boundary color b while reserving flexibility for future goal tracking. The faster that devices cycle, the less flexibility is available in reserve, since devices switching between **Enabled** and **Disabled** become **Refractory**. We thus target a **Flexible** reserve fraction of at least f, meaning that:

$$\frac{|EF_b|}{|ER_b|} \geq f \text{ and } \frac{|DF_b|}{|DR_b|} \geq f$$

Note that these constraints are only one of a number of reasonable possible ways to balance cycling and goal tracking.

This allocation uses the state values expected to hold after the first two allocations are applied, i.e.:

$$|\hat{EF_b}|'' = |\hat{EF_b}| - \Delta_i^{g-} - \Delta_i^{p-}$$

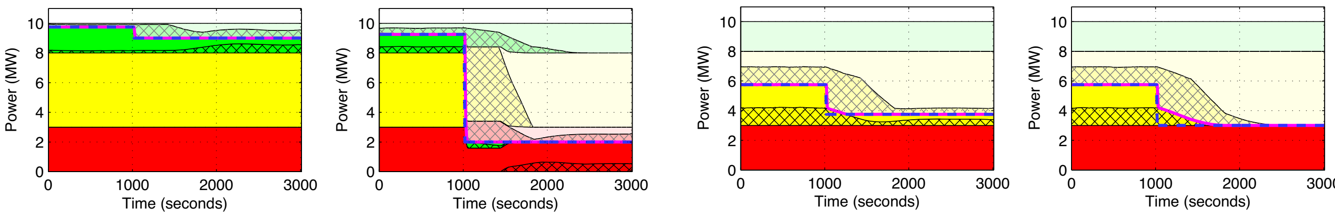
and similarly for each of the other states.

The desired shifts of power, Δ_i^{c-} and Δ_i^{c+}, must be equal for color b (or else goal tracking will be violated), and zero for all colors besides b (or else color priority will be violated). We compute these shifts using the probabilities that will maintain a **Flexible** reserve fraction of f at steady state, $p_{on,ss}$ and $p_{off,ss}$. These probabilities are inexpensive to compute, and can be derived by solving algebraically for equal flux between states. We then compute the power shifts conservatively, using whichever steady state probability leads to a smaller flux given the current states:

$$\Delta_b^{c-} = \Delta_b^{c+} = \min(p_{off,ss} \cdot |\hat{EF_b}|'', p_{on,ss} \cdot |\hat{DF_b}|'')$$

Because the inherent dynamics of the plant that we are controlling are damped, applying the steady-state transition dynamics is guaranteed to eventually lead to steady state.

[2] One might ask: why compute $c(t,a)$ locally at all? The answer is to allow local selection of alternate $c(t,a)$ computations for particular device classes (e.g., compressor scheduling per [11]) with an equivalent expected enablement ratio. Further discussion is beyond this paper's scope.



(a) Case 1: $\Delta \leq F$, $b = b'$ or $R = 0$ (b) Case 2: $\Delta \leq F$, $b < b'$, $R > 0$ (c) Case 3: $F < \Delta \leq F + R'$ (d) Case 4: $F + R' < \Delta$

Figure 3. Simulations on 10,000 devices (1 KW/device, $T_* = 40$ rounds, $f = 1$, $\alpha = 0.8$) illustrating the four cases of step response. Demand is showed as a stacked graph, with **Enabled** demand on the bottom in saturated colors, **Disabled** demand at the top in pale colors, and **Refractory** demand cross-hatched. The goal is a dashed blue line and the current total **Enabled** demand a solid magenta line. When the step is smaller than available flexibility, convergence to the new goal is rapid (a, b); otherwise, convergence time depends on the time for enough **Refractory** demand to return to **Flexible** (c, d).

4) From Allocation Budget to Control: For each color i and each agent a, the values of $p_{on,i,a}$ and $p_{off,i,a}$ are the fraction of allocated versus total flexibility in that color:

$$p_{off,i,a} = \frac{\Delta_i^{g-} + \Delta_i^{p-} + \Delta_i^{c-}}{|EF_i|} \; ; \; p_{on,i,a} = \frac{\Delta_i^{g+} + \Delta_i^{p+} + \Delta_i^{c+}}{|DF_i|}$$

Note that the values do not depend on the identity of the agent, ensuring that the fairness requirement is also fulfilled.

IV. Analysis

We analyze the ColorPower 2.0 algorithm against a key usage scenario: a sudden change in goal. This case is likely to occur in emergencies, such as unexpected failure of a generator or major transmission line. We find ColorPower 2.0 tracks the goal within a small constant factor of optimal performance. Non-emergency scenarios, such as ramps, can then be roughly approximated as a sequence of small steps.

A. Goal Tracking

To analyze step response, consider a goal $g(t)$ equal to some v before time τ and $v - \Delta$ after. We further assume the set of devices is large and homogeneous, such that behavior approximates the continuum limit, and the step occurs when the system is at steady state. We solve for a downward step (e.g., a power transmission failure), assuming α is much larger than zero and that **Refractory** times are many rounds long. The upward step case is precisely complementary.

The step response behavior depends on the amounts of **Flexible** and **Refractory** demand available relative to step size. We will compute the expected behavior, neglecting the small effect of cycling during the main response to the step.

Starting from steady state, the **Flexible** demand available F is all devices in state **EF** from the prior boundary color b to the new boundary color b' and R is **Refractory** demand, which is only from the prior boundary color:

$$F = \sum_{i=b}^{b'} |EF_i| \quad \text{and} \quad R = |ER_b|$$

since all other colors are wholly **Flexible** at steady state. Figure 3 shows the four cases, based on F, R and Δ.

Cases 1,2: If $\Delta \leq F$, then convergence is rapid because the step can be satisfied with available **Flexible** demand. Each round advances an expected fraction α toward the goal, so the rounds r_c to within ϵ of the goal is:

$$\epsilon = (1-\alpha)^{r_c} \cdot \Delta, \text{ where } r_c = \frac{\log \epsilon - \log \Delta}{\log(1-\alpha)}$$

When $b = b'$ or $R = 0$ *(Case 1)*, the step is fulfilled by shifting power from **Enabled** to **Disabled** within a color (Figure 3(a)). When $b < b'$ and $R > 0$ *(Case 2)*, then b' disables more demand than at steady state and is later rebalanced by the color priority allocation (e.g., Figure 3(b)).

Optimal convergence time for this case is a single round, corresponding to $\alpha = 1$, but α should be kept smaller in order to minimize overshoots. For values of α well above zero, the number of rounds to get within any reasonable ϵ remains a small constant, and thus is near-optimal.

Cases 3,4: When $\Delta > F$, convergence time is driven by the time spent waiting for enough **Refractory** demand become **Flexible**, since we assume α well above zero and long **Refractory** time. At steady-state, the distribution of **Refractory** times is piecewise: constant at $\frac{1}{T_{EF,b}+T_{EV,b}/2}$ for the fixed portion of timeout $T_{EF,b}$ and linearly decreasing to zero across the variable range $T_{EV,b}$. Integrating with respect to time gives the cumulative **Refractory** devices becoming **Flexible** over time: a piecewise function linear through $T_{EF,b}$, then rising at a slower quadratic rate. The **Refractory** demand level where this transition takes place is $R\frac{T_{EF,b}}{T_{EF,b}+T_{EV,b}/2}$, which we will denote R' for simplicity.

Thus, if $\Delta - F \leq R'$, meaning demand can be satisfied in the linear range, the expected convergence time is:

$$r_c = \frac{\Delta - F}{R'} \cdot T_{EF,b}$$

Otherwise, demand cannot be satisfied until a portion of the slower range becomes **Flexible**, giving expected time:

$$r_c = T_{EF,b} + \sqrt{\frac{\Delta - F - R'}{R - R'}} \cdot T_{EV,b}$$

Since the limiting factor is the availability of **Flexible** demand, no controller can improve significantly on these times. Thus our controller is near-optimal in all cases.

B. Quiescence Time

The "quiescence time" for the system to return to approximate steady-state gives a conservative bound on how frequently large steps can safely occur. The non-linearities of our controller make closed-form computation difficult, so we analyze it approximately using piecewise estimates.

The number of rounds for quiescence r_q can be conservatively estimated as the sum of the convergence time r_c (calculated in the prior section), the time to satisfy the color priority constraint r_p, and the time for cycling to converge.

Satisfying color priority requires time only if $b < b'$ and $R > 0$. In this case, "extra" flexible demand is drawn from b' to make up for the **Refractory** demand not yet available to disable in b. This will be rebalanced by the color priority allocation after **Refractory** periods elapse in both colors:

$$r_p = \max(T_{DF,b'} + T_{DV,b'}, T_{EF,b} + T_{EV,b})$$

To estimate cycling convergence time, consider the extreme case: all demand was previously in state **EF**, and goal tracking gives rapid convergence (Case 1) to demand being split between **EF** and a "pulse" entering **DR**. Still assuming many **Refractory** rounds, we can coarsely approximate the evolution of this state with the following piecewise elements.

- For the first $T_{DF,b'}$ rounds, no demand shifts state, as the pulse in **DR** progress through **Refractory** time.
- By $T_{DF,b'} + T_{DV,b'}$ rounds all of the original pulse has become flexible, overshooting the desired reserve f. Thus, at some point in the last $T_{DV,b'}$ rounds, the rate of demand exchanged between **Enabled** and **Disabled** has crossed the steady-state rate. From this point onward, the rate of cycling is relatively close to steady-state.
- After another $T_{EF,b'} + T_{EV,b'}/2$ rounds, cycling results in near-steady-state allocations in all the **ER** states.

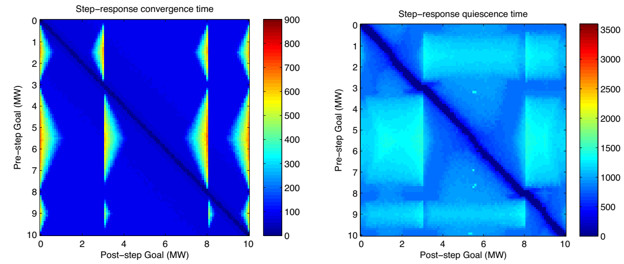
We may thus conservatively estimate that two full cycles of expected **Refractory** periods should suffice to bring the system close to steady state. The length of these cycles depends on the position of the goal within the boundary color and the size of the flexible reserve. All told, therefore, quiescence time following a step is expected to be less than:

$$r_q = r_c + r_p + 2(T_{DF,b'} + T_{DV,b'}/2 + T_{EF,b'} + T_{EV,b'}/2)$$

Assuming similar timeouts for b and b', this is once again within a small constant factor of optimal, since it is impossible for any controller to reach quiescence faster than one full set of expected **Refractory** timeouts, as no shorter time can distribute devices through state **ER**.

V. Empirical Results

We validated our analytic results in simulation, also showing scalability and resilience to goal change and inaccuracy. Our simulations were implemented in MATLAB, using the assumptions stated in Section II-B. Except where noted otherwise, all simulations are executed with the following parameters (which are the same as for Figure 3): 10 trials per condition for 10,000 controllable devices, each device consumes 1 KW of power (for a total of 10 MW demand), devices are 20% green, 50% yellow and 30% red, measurement error is $\epsilon = 0.1\%$ (0.001), rounds are 10 seconds long, all **Refractory** time variables are $T_* = 40$ rounds, the flexible reserve ratio is $f = 1$, and the proportional control constant is $\alpha = 0.8$. Error is measured by taking the ratio of the difference of a state from optimal versus the total power.



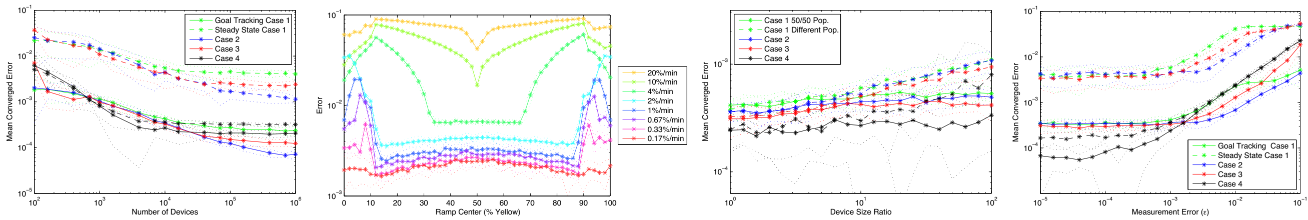
(a) Measured Convergence Time (b) Measured Quiescence Time

Figure 4. Following a step in goal, measured convergence time closely matches predictions; quiescence is much better than the conservative bound.

To validate our convergence and quiescence time predictions, we consider steps where the goal before and after the step ranges from 0 to 10 MW in increments of 0.1 MW, executing one trial per condition. Simulation begins in steady state and proceeds for 3600 seconds following the step. We consider the controller to have *converged* to the goal when the **Enabled** demand never again departs by more than 1% (0.1 MW) average over a minute, and to have reached *quiescence* when it never departs from the steady state distribution of demand into states by more than 2% (0.2 MW). Figure 3 shows examples of these simulations.

Figure 4(a) shows the measured convergence time, showing a tight conformance with predictions: fast convergence except when waiting for **Refractory** power, which happens only when closely approaching the boundary of a color without crossing into the next color. Quiescence time (Figure 4(b)) also matches predctions, and is much faster than the conservative estimates derived in Section IV-B—likely due to the r_q estimate being overly conservative.

The stochastic nature of the controller makes it highly scalable, and the law of large numbers means control should improve as the number of devices increases. Figure 5(a) compares convergence and quiescence error for systems of 100 to 1,000,000 devices, simulated for the four step cases shown in Figure 3 with the error averaged over the last 10 simulated minutes (60 rounds). Goal tracking is extremely accurate, down to a limit imposed by the accuracy of the estimator (ϵ), and degrading gracefully for fewer devices as predicted by binomial variance. There is greater difference between the steady state predictions and the observed **Flexible** versus **Refractory** distribution, since error integrates



(a) Scalability (b) Ramp Error (c) Heterogeneous Demand (d) Estimation Error

Figure 5. Accuracy improves with larger numbers of devices (a), to the limit imposed by estimator accuracy ϵ. Ramps (b) have higher accuracy in goal tracking when the slope is lower and the ramp does not come close to fully enabling or disabling a color. Increasing heterogeneity in the amount of demand per device (c) or increasing error in demand estimates (d) leads to a slow graceful degradation of controller performance. Dotted lines show ± 2 std. dev.

over the **Refractory** period. (The exception is in case 4 of Figure 3 where steady state after the step has no **Refractory** power.) The error is still sufficiently small, however, and follows the same trends as for goal tracking.

We evaluate ramp response by considering saw-toothed waves of goals, with means at various locations in the yellow demand range, an amplitude of ± 0.5 MW, and a ramp duration of 60 to 7200 seconds. The controller is run for 6 simulated hours and the results are drawn from the last 4 hours. Figure 5(b) shows the maximum error at any point during the simulation, as a function of both the mean goal and the ramp rate. The error is symmetric and bimodal: when the ramp duration approaches the expected **Refractory** time and the goal is not near the limits of the color, flexible power can be restored as quickly as ramping consumes it and the error is low. When the ramp is fast or closely approaches the limits of the color, however, device cycling cannot keep up with the changing goal, and error increases.

The stochastic nature of the controller means it should degrade gracefully in the face of inaccuracies such as heterogeneity or measurement error. To test heterogeneity, we break devices into two populations with a demand ratio from 1 to 100 and the same 1KW mean demand, with either equal numbers or equal ratios above and below 1KW. Figure 5(c) shows results for simulation of the four step cases of Figure 3, reporting error as for the step case. To measure impact of estimation error, we range ϵ logarithmically from 10^{-5} to 10^{-1}. Figure 5(d) shows results for the four step cases of Figure 3, reporting error as before. In all cases, error increases gradually with inaccuracy, showing graceful degradation and good behavior even with high inaccuracy.

VI. Contributions

We have presented the ColorPower 2.0 algorithm for fast, precise distributed shaping of demand. The algorithm is near-optimal in its convergence and quiescence time, and its stochastic basis means that the larger the network the better the expected performance. Simulations confirm this analysis, showing fast, effective control for networks with around 1000 devices or more, and a high degree of tolerance for shifting demand, device heterogeneity, and measurement error. Future work includes tightening of the controller performance and analytic bounds thereon, e.g., through inclusion of future predictions rather than just instantaneous goals. More importantly, however, through industry partnership we expect to soon begin pilot deployments of ColorPower, and expect to focus on pragmatics required for those deployments.

References

[1] V. Ranade and J. Beal, “Distributed control for small customer energy demand management,” in *IEEE SASO 2010*, 2010.

[2] Office of Energy Statistics, “Annual energy review,” U.S. Energy Info. Admin., Tech. Rep. DOE/EIA-0384(2010), 2011.

[3] S. Darby, “The effectiveness of feedback on energy consumption. a review for defra of the literature on metering, billing and direct displays.” Environmental Change Institute, University of Oxford, Tech. Rep., 2006.

[4] D. Hammerstrom, T. Oliver, R. Melton, and R. Ambrosio, “Standardization of a hierarchical transactive control system,” in *Grid Interoperability 2009 Conference*, November 2009.

[5] Comverge Inc., “New demand response programs offer big cash payments to business and industry,” Whitepaper, Comverge website, 2010.

[6] EnerNOC Inc., “The Xcel Energy-EnerNOC Peak Savings Program,” EnerNOC Website, 2008.

[7] C. Benes, R. Premus, M. Ruth, C. Stimmel, A. Tuck, and A. Wheeler, “Consumer directed energy management systems and methods,” U.S. Patent App. 12/555,751, September, 2009.

[8] E. Ogston, A. Zeman, M. Prokopenko, and G. James, “Clustering distributed energy resources for large-scale demand management,” in *IEEE SASO 2007*, 2007.

[9] R. Kulyk and M. Kerbel, “Method and apparatus for implementing enablement state decision for energy,” U.S. Patent Application No. 7,580,775, August, 2009.

[10] V. V. Ranade, “Model and control for cooperative energy management,” Master’s thesis, MIT, June 2010.

[11] A. Zeman, M. Prokopenko, Y. Guo, and R. Li, “Adaptive control of distributed energy management: A comparative study,” in *IEEE SASO 2008*, 2008.