Disaggregating Multi-State Appliances from Smart Meter Data

Daniel "Jack" Kelly and William Knottenbelt
Department of Computing
Imperial College London
South Kensington Campus
London SW7 2AZ
{dk3810, wjk}@doc.ic.ac.uk

Smart electricity meters record the aggregate consumption of an entire building. However, appliance-level information is more useful than aggregate data for a variety of purposes including energy management and load forecasting. Disaggregation aims to decompose an aggregate signal into appliance-by-appliance information.

Existing disaggregation systems tend to perform well for single-state appliances like toasters but perform less well for multi-state appliances like dish washers and tumble driers.

In this paper, we propose an expressive probabilistic graphical modelling framework with two main design aims: 1) to represent and disaggregate multi-state appliances and 2) to use as many features from the smart meter signal as possible to maximise disaggregation performance.

1 Introduction

Research on consumer behaviour indicates that people are better able to manage their energy consumption when given disaggregated, appliance-by-appliance information instead of aggregate information alone [3]. For example, an individual might like to know how much energy their fridge uses so they can work out whether it would be cost effective to replace it with a more efficient version.

How can appliance-by-appliance information be provided to the maximum number of users, whilst requiring the minimum effort per user?

Aggregate data will soon be commonplace. The UK government requires that every house should have a smart meter installed by 2019[1]. The draft specification for smart meters states that meters should report readings to the "home area network" once every five seconds[2]. It would be very useful if this smart meter data could be accurately disaggregated in software.

Research into disaggregation started in the mid-1980s [5] and has become especially active in the past few years due in part to high energy prices. Modern state-of-the-art disaggregation algorithms are capable of handling single-state appliances like toasters with high accuracy. So, what is preventing the mass adoption of disaggregation systems?

2 Challenges

There are several challenges still to overcome. Two overlapping sets of appliances which cause problems for disaggregation algorithms are: 1) multi-state appliances (like washing machines) and 2) appliances which emit complex, rapidly changing waveforms (like tumble driers).

Submitted to:
ICEP 2012

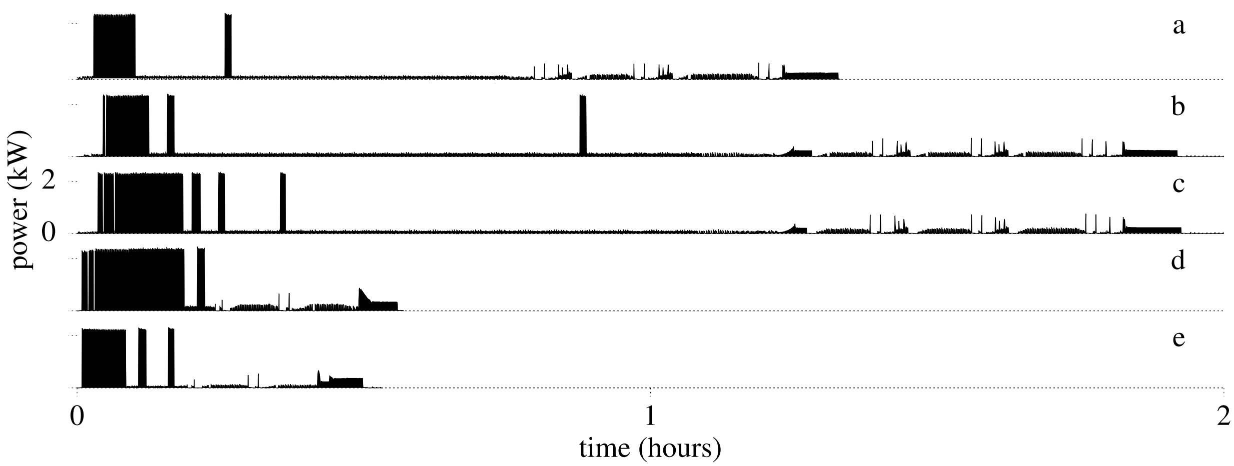

Figure 1: Five runs of the same washing machine. Runs *a,b* & *c* followed the standard washing program. Runs *d* & *e* were the quick wash program. The 2 kW spikes indicate when the washer uses its heater.

2.1 Multi-state appliances

Figure 1 shows five runs of the same washing machine. Not only can it exist in multiple states, but the transition between states varies from run to run.

2.2 Rapidly changing appliance waveforms

Figure 2 shows the power consumed by a tumble drier, sampled at 1 Hz. Note the low-amplitude ($\approx$ 200 W) repetitive spikes during the first forty minutes (these correspond to the drum spinning one direction for about one minute; stopping; and reversing for another minute etc). Then, after forty minutes, the heating element cycles on and off to prevent the drier from exceeding some maximum temperature; a behaviour which produces high-amplitude ($\approx$ 2 kW) spikes in the power consumption.

Why are these rapid changes a problem? Consider that the first step taken by many disaggregation algorithms designed to work on standard smart meters is to simplify the aggregate signal to a sequence of *steady-states*[6]. Steady-state algorithms have two options when faced with an appliance like a tumble drier: either smooth out the rapid changes (hence losing a lot of information: those high frequency features are rather distinctive) or attempt to track the high frequency changes (hence violating the design assumption that a steady state is *steady*). A further issue with rapidly-changing waveforms is that the smart meter may sample the aggregate signal at sub-Nyquist rates, hence aliasing the signal.

3 Our proposed solution

There is one fundamental point of divergence between our proposed solution and existing solutions: our approach does not start by finding steady-states, instead we want to make use of as much information in the aggregate signal as possible. We want to treat bizarre waveforms and complex state sequences as distinctive features which can be used to maximise the performance of the disaggregation algorithm.

This fundamental difference requires that we build a probabilistic modelling framework capable of capturing the rules which govern both the complex waveforms emitted by each state and the state transitions. Instead of modelling appliances as "black boxes", we plan to model the main internal components

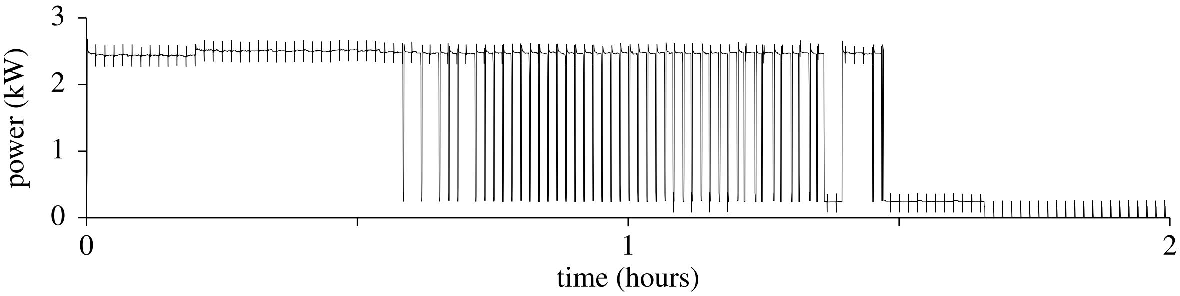

Figure 2: Tumble drier signature sampled at 1 Hz

of each appliance, thereby allowing us to build faithful, expressive models of appliances. Our framework is hierarchical: the bottom layer models individual components, the next layer up represents entire appliances and the top layer represents interactions between appliances and the wider environment. The bottom layer will be hard-coded; the upper two layers will be learnt from aggregate data.

3.1 Bottom layer: parametrised component models

All appliances are constructed from a set of *components* such as motors, heaters and compressors [1]. This set of components is far smaller than the set of all appliances. We estimate that the majority of domestic appliances could be modelled using a single "component vocabulary" of approximately five components.

Different components produce different waveforms. Components will be modelled using simple mathematical formulae. For example, the power consumed by a heater decays over time in a relationship which can be modelled by $p = \frac{a}{t+b} + c$ where p is power, t is time and a, b and c are constant parameters (preliminary experiments fitting this model to toaster data achieve $R^2 > 0.99$).

3.2 Middle layer: probabilistic graphical models of appliances

Component models will be combined into probabilistic graphical *appliance models* which are similar to a Markov chain. Each node will describe the state (*on*, *off*, *cycling* or *ramping*) of every component of the appliance. The time duration of each state will be explicitly represented.

3.3 Top layer: inter-appliance relationships

The top level of our model will capture relationships between appliances (e.g. if the games console is on then the TV will probably be on too[4]); relationships between appliances and the time of day; and we will experiment with modelling hidden parameters such as house occupancy (some appliances require manual operation, others function automatically).

3.4 Disaggregation

Once the appliance models are built, how do we use those models to disaggregate an aggregate signal? Our disaggregation procedure will run in two phases: first get a rough estimate of when each appliance

[1]The power consumption of highly complex appliances like computers is likely to be modelled by a single random variable with a specified probability distribution, although we would like to investigate more detailed modelling of computers.

is active by searching for distinctive *features* encoded by each model; then use the current estimate to reconstruct the full appliance signature, match the reconstruction to the aggregate signal and then iteratively improve the fit of the model to the aggregate signal (hence evaluating the fit of the model to the data using *every* data sample within the search window rather than a sequence of steady-states [2]).

4 Conclusion

Our system should be capable of reconstructing realistic waveforms for any appliance. Why might this be worthwhile? Our system should be able to take advantage of quirky features of the aggregate signal to maximise recognition performance and fit the appliance models very tightly to the aggregate signal to achieve a good estimate of total energy consumed.

5 Next steps

Implement the disaggregation algorithm described above and evaluate the performance of our system using our own smart meter data and data from MIT's Reference Energy Disaggregation Data set[7].

References

[1] DECC & Ofgem (2010): *Smart meter Implementation Strategy Prospectus*. Technical Report, DECC, Ofgem/Ofgem E-Serve. Available at `http://www.ofgem.gov.uk/e-serve/sm/Documentation/Documents1/Smart%20metering%20-%20Prospectus.pdf`.

[2] DECC & Ofgem (2011): *Design requirement: Annex - Functional catalogue - Smart Metering Implementation Programme*. Technical Report. Available at `http://www.decc.gov.uk/assets/decc/Consultations/smart-meter-imp-prospectus/1480-design-requirement-annex.pdf`.

[3] Corinna Fischer (2008): *Feedback on household electricity consumption: a tool for saving energy?* Energy Efficiency 1(1), pp. 79–104, doi:10.1007/s12053-008-9009-7.

[4] H. Kim, M. Marwah, M. F. Arlitt, G. Lyon & J. Han (2011): *Unsupervised Disaggregation of Low Frequency Power Measurements*. In: 11th International Conference on Data Mining, Arizona, pp. 747–758. Available at `http://www.cs.uiuc.edu/homes/hanj/pdf/sdm11_hkim.pdf`.

[5] G. W Hart (1984): *Nonintrusive Appliance Load Data Acquisition Method*. Technical Report, MIT Energy Laboratory Technical Report.

[6] G. W Hart (1992): *Nonintrusive appliance load monitoring*. Proceedings of the IEEE 80(12), pp. 1870–1891, doi:10.1109/5.192069. Available at `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=192069`.

[7] J. Zico Kolter & Matthew J. Johnson (2011): *REDD: A public data set for energy disaggregation research*. In: SustKDD workshop, San Diego, California. Available at `http://redd.csail.mit.edu/kolter-kddsust11.pdf`.

[8] R.A. Newcombe, S.J. Lovegrove & A.J. Davison (2011): *DTAM: Dense tracking and mapping in real-time*. In: Computer Vision (ICCV), 2011 IEEE International Conference on, pp. 2320 –2327, doi:10.1109/ICCV.2011.6126513.

[2] This approach of bootstrapping the search algorithm by first searching for *features* and then switching to a dense, full-sample-rate reconstruction is similar in spirit to a state of the art technique in computer vision known as Dense Tracking and Mapping (DTAM)[8].