Testing Applicability of Virtual Stochastic Sensors for Non-Intrusive Appliance Load Monitoring

Claudia Krull [1]   Marcus Thiel [2]   Graham Horton [3]

Faculty of Computer Science
Otto-von-Guericke-University
Magdeburg, Germany

Abstract

Non-intrusive appliance load monitoring (NIALM) aims at reconstructing the electricity consumption of household appliances based only on the cumulative consumption data collected via a smart meter. Various approaches have been proposed to perform NIALM including various types of Hidden Markov Model (HMM) offspring. Most do not consider the explicit duration of an appliance activation. Virtual stochastic sensors (VSS) and their underlying Hidden non-Markovian Models (HnMM) can include explicit process durations. This paper tests whether VSS can solve the NIALM task, and analyzes the methods reconstruction accuracy on the publicly available SMART* data set. Models of the household appliances with different inherent states are automatically extracted from the appliance data. The combined model, including a subset of the appliances, is then used to disaggregate the cumulative energy consumption data. Experiments show a reconstruction accuracy of up to 90% with appropriate method parameters, showing that VSS can compete with existing NIALM approaches.

Keywords: Virtual Stochastic Sensors, Hidden non-Markovian Models, Non-Intrusive Appliance Load Monitoring

1   Introduction

Declining reserves in fossil fuels and an increasing energy demand put a strain on the future energy market. Developing alternative energy sources is one way of countering that problem. Another approach, that should not be disregarded is the potential in energy saving, not just in industry, but also in private households. In order to do that, consumers first need to be aware of their consumption patterns.

Smart meters make it easily possible to closely monitor the energy consumption on a household basis. There is however no easy or cheap way to directly monitor

the electricity consumption of particular appliances. Non-intrusive appliance load monitoring (NIALM) is the research field concerned with dis-aggregating this cumulative measurement in order to reconstruct the particular appliances share. This information can help consumers change their consumption systematically.

NIALM approaches build models of a households appliance set by supervised or unsupervised training, which can then subsequently be used to dis-aggregate future measurements of the smart meter. Most models do not consider explicit state durations for the appliance states, or need complicated extensions to handle explicit state durations. Some also cannot accommodate for more than two possible appliance states (usually only on- and off-states), which is often a too crude representation for some major appliances such as for example a washing machine, a dryer or a heater. [20,19]

Virtual stochastic sensors (VSS) were built to reconstruct hard-to-obtain measures of interest from measurable quantities, where the relationship is not a closed analytical expression, but can be represented by a discrete stochastic model, e.g. a Hidden non-Markovian Model (HnMM). These models can by construction represent non-Markovian state holding times, arbitrary states numbers and multiple parallel processes. A Masters thesis was conducted in order to test the applicability of VSS and HnMM to NIALM. [23] We were able to build and apply models with up to three states per appliance and non-Markovian state holding times, reaching competitive reconstruction accuracy. This paper summarizes the thesis and points to future research to provide a general approach using VSS for NIALM.

The next section will present related and previous work on non-intrusive appliance load monitoring and on virtual stochastic sensors in the next section. Then we elaborate our case study of NIALM through VSS on the SMART* data set. In the experiment section we evaluate the capabilities and limitations of the approach, followed by conclusions and a outlook.

2 Related and Previous Work

2.1 NIALM Datasets

Several data sets are publicly available with different properties, all of them containing cumulative as well as appliance level energy consumption data.

The UMass Smart* Data Set for Sustainability (2013 release) was collected over an extended period of time in three private homes. It includes not only the aggregate power readings, but also circuit and appliance level energy consumption information, as well as environmental data. This makes it uniquely suitable for supervised training approaches. [2]

The BLUED data set (Building-Level Fully-Labeled Electricity Disaggregation Dataset) contains data sampled at a very high frequency (12 kHz) over a whole week in a home in the USA. The data set is fully labeled with appliance state transitions and time stamps in order to enable supervised learning approaches and evaluating the quality of energy disaggregation methods by comparing the results to this ground truth. [1]

The REDD data set (Reference Energy Disaggregation Data Set) also contains data collected from several private homes on a circuit level, as well as the cumulative data. It was collected having existing approaches and their differing requirements in mind. However, the data does not contain as much information on specific devices, making it less interesting for the work at hand. [12]

We have selected to use the Smart* data set for this work. More specifically, we chose to test the reconstruction on circuit level data from House A. One reason is, that our approach does not need high frequency samples, but longer traces, which rules out the BLUED data set. Compared to the REDD data set, the data is more detailed, which led to the decision. The REDD data set will however be used in future experiments as a further benchmark.

2.2 Existing NIALM Approaches

Extensive research has been done in recent years with various methods being applied to NIALM, with varying features, goals and results, some of which are mentioned here. Baranski and Voss [3] introduce a method using fuzzy clustering, a genetic algorithm and dynamic programming to dis-aggregate a low frequency data stream into appliances with possibly multiple states. Zeifmann et al. [25,24] show and compare different methods to directly identify appliances and decompose the cumulative energy consumption into its components, without a graphical model such as an HMM. Kang et al. [10] use optimization methods to dis-aggregate the cumulative power reading, namely mixed integer pattern optimization and linear programming, both with good results.

Several NIALM approaches use Hidden Markov Models (HMM) and related model types to model the household energy consumption by appliance. Kim et al. [11] use different variants of Conditional Factorial Hidden Semi Markov Models to represent the household appliance set, with gamma and exponential distributions for the state durations. In contrast, we consider not just the on-off states of an appliance, but try to train models with possibly several distinct states. Kolter and Jaakkola [13] apply additive Factorial HMM to energy disaggregation, restricting the inference algorithm in order to make the problem computationally tractable. Nylund [19] uses a Gibbs sampling algorithm and Gamma distributions to represent explicit state durations, does however not include appliance models with more than two states. Parson et al. [21] develop an unsupervised learning approach, which uses general appliance models and tunes them to a particular households cumulative measurement. Parson [20] also shows that HMM cannot accurately represent some appliance constellations, but HSMM can, showing the need for explicit state durations. He also learns models with more than two states from the data, depending on the type of appliance.

In some of these graphical modeling approaches the appliances are represented by their on- and off-states and possibly their durations. Most models use factorial structures to combine the power consumption of the appliances in the states to a cumulative value, which represents the data measured by the smart meter. However, none of the models allows for arbitrary distributions for state durations, and at the

same time considers more than two states per appliance. This is where we hope to improve by using the virtual stochastic sensors and Hidden non-Markovian Models.

2.3   Virtual Stochastic Sensors

Virtual stochastic sensors [15] are a method to help gain insights into partially observable discrete stochastic systems. The modeling paradigm employed are Hidden non-Markovian Models (HnMM), which extend Hidden Markov Models (HMM) [22,8] by allowing arbitrary continuous distribution functions for activity durations. [16] The evaluation task of finding the probability of an observed output sequence (trace) and the decoding task of finding a likely system behavior (path) to have caused an observed output sequence (trace) can be solved for HnMM. The solution approach employing the Proxel analysis method [9,18] is briefly described here. For Conversive HnMM, a subclass of HnMM, unsupervised training approaches can also be realized. [5] Beyond several academic examples [14,15], VSS have been successfully applied to reconstructing item paths from RFID sensor readings in job-shop environments [17] and for gesture recognition on multi-touch devices [7,6].

   Applying VSS to the problem of non-intrusive appliance load monitoring seems promising for several reasons. HMM and several related modeling paradigms have been successfully applied to NIALM. HnMM enable the modeling of explicit state holding times, which was shown to improve accuracy for NIALM. Furthermore, our method does not inherently restrict the number of states per appliance, making realistic load modeling and reconstruction possible.

2.3.1   VSS Behavior Reconstruction via Decoding using HnMM and Proxels

In the specific case of NIALM, the Decoding task is of interest, where the behavior reconstruction tries to determine a likely system behavior (path) to have produced a give system output protocol (trace). The Decoding procedure for HnMM is elaborated in [14].

   Virtual Stochastic Sensors use Augmented Stochastic Petri nets (ASPN) [5] as user models to represent the doubly stochastic process consisting of the stochastic system behavior and its stochastic relationship to the observable output. An ASPN can be converted into a Hidden non-Markovian Model (HnMM) [16] as computational model, which represents the model as its state space with respective outputs.

   The Proxel method is then employed as solution algorithm to perform the actual behavior reconstruction. The method generates the reachable system state space in discrete time steps in the form of probability elements (Proxels). Each Proxel contains information on the discrete system state dS, the age of all non-Markovian transitions τ, the system time t, the route or path by which it was reached R and the probability of that combination p. In each step, the successor Proxels are determined by the possible firings of the currently enabled state transitions and the possibility of staying in the same state. The probabilities of these system developments are calculated using the hazard rate function (HRF). [9,18]

   When using Proxels for VSS analysis, the generated possible system developments are compared to the observed output trace. Only the Proxels with paths R

that could have produced the given trace are kept in the algorithm and weighted with the output probability of the observed symbols.

As common for state-space based analysis methods, the Proxel method can suffer from severe state space explosion. This can be countered by merging Proxels with the same dS, τ, R and by discarding Proxels below a minimum probability threshold. However, in behavior reconstruction, the development path itself is of interest and merging Proxels with different paths is no longer possible. This leaves thresholds as the only measure to counter the state space explosion, which need to be carefully tuned, not to lead to a complete dying out of the reachable state space.

The next section describes the concrete adaptions needed when applying VSS with Proxels and HnMM to NIALM.

3 Applying VSS to NIALM

As a first step to a behavior reconstruction, we need to construct feasible appliance models, which can then be combine in a joint model for later reconstruction of the individual appliance loads.

3.1 Construction of Appliance Models

The Smart* data set contains circuit level energy consumption data, where a circuit can represent a single appliance (e.g. *Dryer*, *DuctHeater*) or a set of appliances (e.g. *Master Outlets*, *Kitchen Lights*). Both will be treated the same and used synonymously in the following.

We used this circuit level data to construct the single appliance models, shown here for the *Dryer* appliance. For a realistic load reconstruction, we first identified distinct appliance (consumption) states to later mimic the actual progression of these states in the actual HnMM. Figure 1 shows an excerpt of the protocol of the energy consumption of the *Dryer*, which seems to show at least three distinct operational states, including off/standby, with different energy consumption.

Initially we manually identified five distinct consumption states with different energy level ranges, which we dubbed *Standby*, *Z1*, *Z2*, *Z3* and *Z4*. We then analyzed the consumption data for this appliance and extracted samples for the states duration, and the frequency of switches between specific states. The state duration samples were then fitted to a Weibull distribution and the followup state probabilities computed. The distribution of the different consumption levels within one state was recorded in Histograms, shown in Figure 3.

We then constructed the ASPN depicted in Figure 2. The model contains ten places representing five tangible states, and 5 vanishing states, both represented as circles. The tangible states *Standby*, *Z1*, *Z2*, *Z3* and *Z4* represent the actual appliance states. They are followed by open rectangles denoting the tangible states durations, which are elaborated in Table 1. The vanishing states *!Standby*, *!Z1*, *!Z2*, *!Z3*, *!Z4* are used to model the conditional transition from one tangible state to the next. They are followed by solid rectangles modeling immediate state transitions and are annotated with the transition probabilities from a state to its successors.
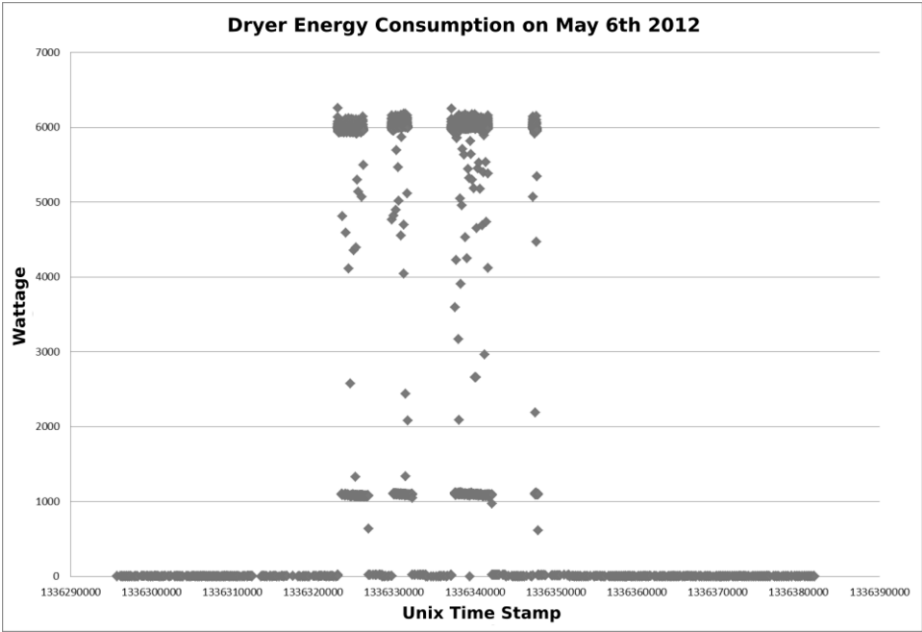
Fig. 1. Excerpt of Energy Consumption (Watt) Protocol of the Dryer Appliance

The tangible states are further annotated with emissions, representing the energy consumption behavior in these states. The ranges are only placeholders for a more fine granular representation using histograms for each states consumption, represented in Figure 3.

State	Duration Distribution	Mean Duration
Standby	Weibull(0.5830, 189.9696)	297.04
State Z1	Weibull(0.5096, 156.5068)	302.40
State Z2	Weibull(0.6046, 771.5454)	1149.44
State Z3	Weibull(1.3207, 184.9010)	170.24
State Z4	Weibull(1.3202, 159.0938)	146.49

Table 1
State Durations of Dryer HnMM

Having determined a suitable model structure, a method was developed to automatically extract these models from the recorded appliance data. The only input needed for the automatic model construction was a manually determined set of appliance states with their respective minimum and maximum load. The probabilities, distributions and histograms are then extracted from the appliance consumption data and the model parametrized automatically.
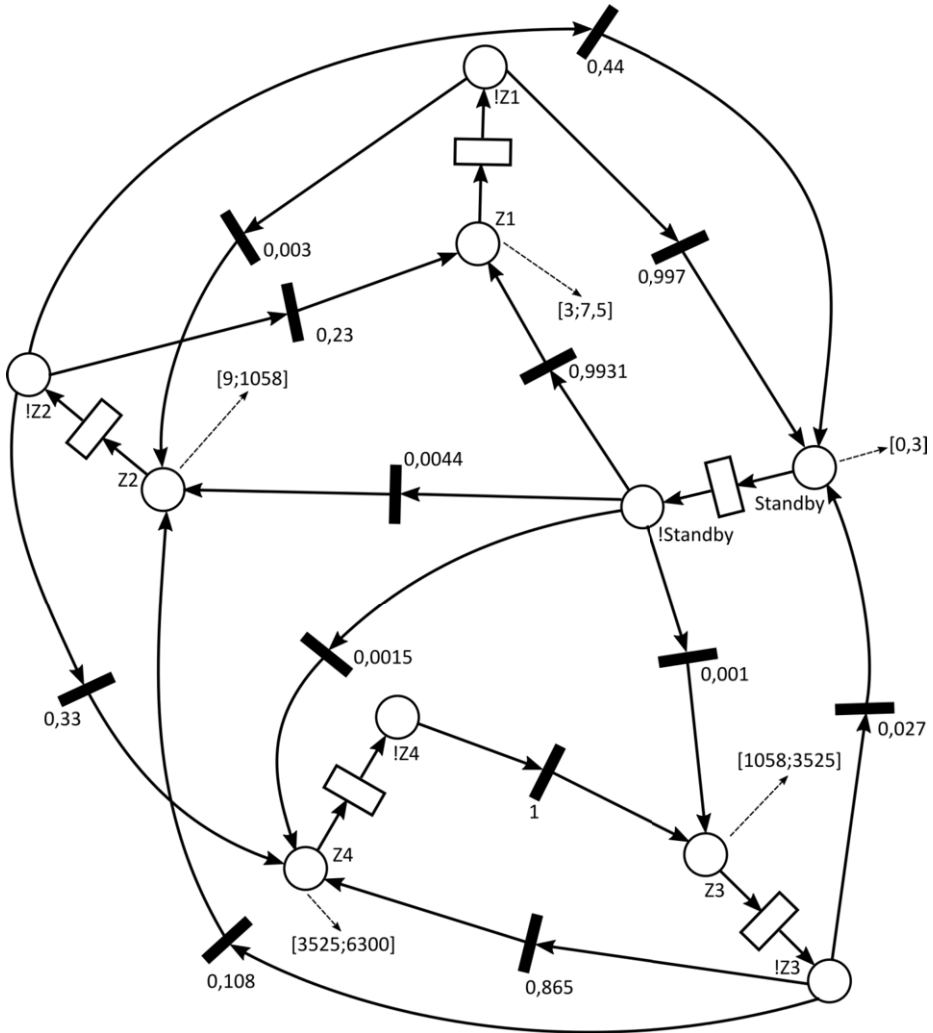
Fig. 2. Dryer Appliance ASPN

3.2 Joint Appliance Set Model

The single appliance ASPNs were combined into one ASPN representing the complete set of household appliances with their usage states and respective energy consumption behavior. Each appliance model contained one token in the place representing the current state of the appliance. The state space of the model is then the set of currently active states of the different appliances, represented in the marking of the ASPN through the distribution of the tokens.

The emissions of the currently active tangible appliance states were summed up to result in a cumulative energy consumption. This combined model contained 25 consumers or circuits with three to seven states each, resulting in over $3E11$ possible discrete model states. This model was unfortunately too large, both in memory consumption and computational effort to process even one step of the Decoding algorithm.
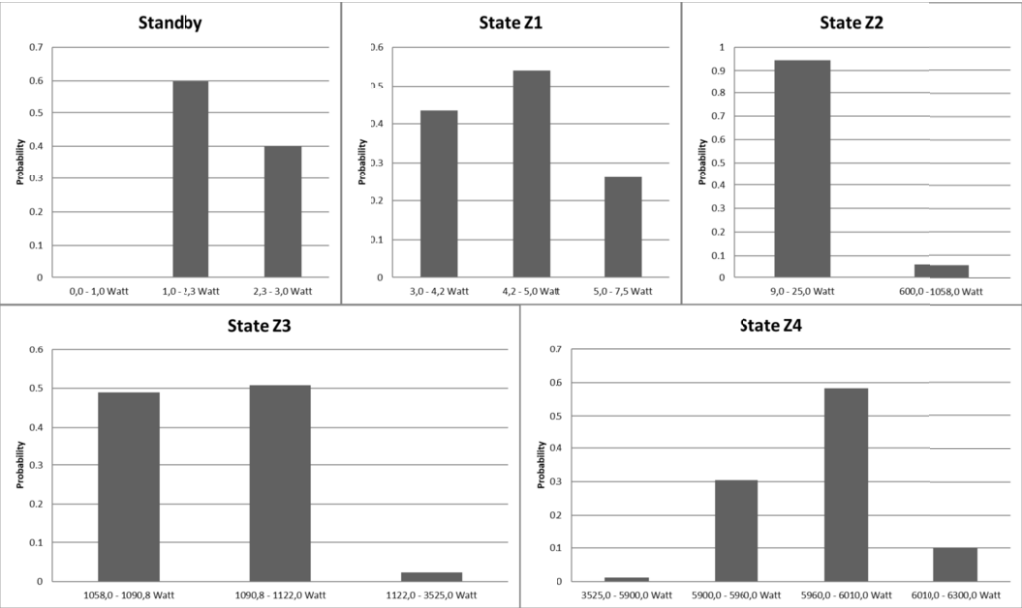
Fig. 3. Dryer Appliance State Energy Usage Histograms

In order to reduce the model state space to a feasible size, two reduction methods were used. First the number of appliances in the model itself was reduced to the 11 most frequently used circuits (*Counter Outlets 1* and *2*, *Dishwasher*, *Dryer*, *Duct Heater*, *Furnace*, *Kitchen Lights*, *Living Room Outlets*, *Master Lights* and *Master Outlets*), since these provided more data points and had more impact on the overall consumption. The remaining circuits data was combined into a twelfth unknown consumer, using the model construction technique described in Section 3.1 on their combined data points. Thus the analysis could still be conducted on the original cumulative consumption data, and no artificial cumulative consumption had to be constructed. The second reduction was to reduce the appliance models to two states each, three in the case of the *Dishwasher* and *Dryer*, since these showed three very distinct consumption states. Using these reduces state sets as input for the automated model construction, new appliance ASPNs were constructed, using the process described in Section 3.1. This unfortunately again limited the realism of the appliance representation, however, the state duration is still represented by a Weibull distribution.

These measures resulted in a drastic reduction of the combined model state space to only 4 608 discrete states, entailing a reduction of memory requirement and computational effort to feasible regions.

This reduction in model accuracy of course also causes a loss in reconstruction accuracy. Qualitatively, the differentiation of StandBy and Off states was no longer possible for two-state appliances and the removal of over half of the appliances of course prevents the reconstruction of the consumption caused by them. A quantitative evaluation of the loss in modeling accuracy is currently not easily possible, since current research is still aiming to identify a measure for model accuracy and

an estimate for possible reconstruction for Hidden non-Markovian Models. The experiments in Section 4 will evaluate the reconstruction accuracy which can be achieved using this reduced model.

3.3 Decoding for NIALM

The decoding task is to determine the most likely state sequence (path) to explain a given observations sequence (trace), in our case a cumulative energy consumption. The Decoding was conducted as described in Section 2.3 and in [14]. Irregular measurement intervals in the appliance data made the assumption necessary that the appliance consumption does not change in between consecutive measurements.

For testing purposes, three appliances from the whole set were selected and their ASPNs combined to form a combined household appliance model: *Dryer*, *Master Lights* and *Counter Outlets 1*. The trace used for reconstruction was artificially created by combining the consumption data of the three appliances, simultaneously yielding a ground truth.

The path reconstruction was evaluated by comparing the reconstructed appliance states in the Proxel route R with this ground truth in two different ways:

- *Measure 1* compares the reconstructed appliance state with the ground truth state only in time steps when there is a recorded consumption data point. See Equation 1, where CD is the number of correctly classified appliance states in time steps with recorded consumption data, and d is the total number of consumption data points for this appliance. This places more weight on consumers with many measurements, regardless of whether these are due to noise or frequent consumption changes.

- *Measure 2* compares the reconstructed appliance state with the ground truth state in all time steps, assuming no changes in the ground truth appliance state in between consecutive measurements. See Equation 2, where CT is the number of correctly classified appliance states in all time steps, and n is the total number of time steps in the trace.

$$M1 = \frac{CD}{d} \tag{1}$$

$$M2 = \frac{CT}{n} \tag{2}$$

The accuracy of the path reconstruction was 0.95 for *measure 1* and 0.78 for *measure 2*. The deterioration in Version 2 being solely due to a degradation in reconstruction accuracy of the *Dryer* states. Overall, this test demonstrated that recontructing energy consumption from cumulative data using VSS is possible.

4   Experiments

In the experiments we first determined the effect of two crucial Decoding algorithm parameters, the pruning threshold, which determines the number of Proxels retained in every time step and the trace length, which represents the time interval for
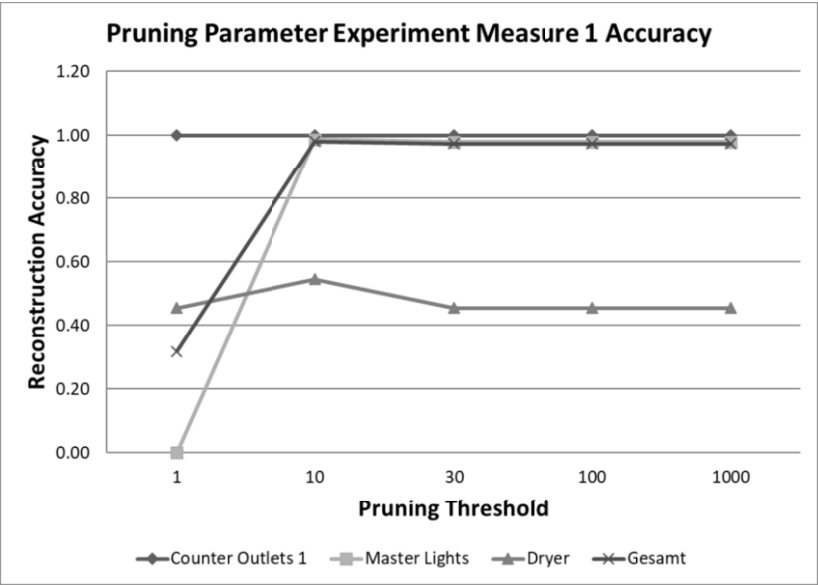
Fig. 4. Influence of Pruning on Reconstruction Accuracy Measure 1 Overall and for Single Appliances

which the reconstruction is performed. These tuning experiments led to a choice of parameter combination for the more elaborate reconstruction accuracy experiments.

For the experiments we selected different days for training and evaluating our models from the circuit level data of House A from the SMART* data set. [2]

4.1   Tuning Experiments - Effect of Pruning and Trace Length

In the tuning experiment, the three-consumer model from Section 3.3 with the original state configuration was used for faster results. As data we used the appliance and network consumption from or starting May 6th, 2012. The measure of interest was the reconstruction accuracy determined using measure 1 (only time steps with measurements are compared) and measure 2 (all time steps are compared).

Figure 4 shows the effect of the pruning threshold, which is varied from 1 to 1000, on the accuracy *measure 1*. The graph shows, that increasing the pruning threshold to retain more Proxels per time step increases the reconstruction accuracy initially, but the effect diminishes with larger thresholds. Retaining only the one most likely Proxel in every step is obviously not feasible, since this allows the reachable state space to die out, should the one remaining path become impossible when compared to the trace. Retaining 10 to 30 Proxels yielded good results, therefore, a pruning threshold of 30 was selected for the performance experiment.

Figure 5 shows the influence of the input trace length on reconstruction accuracy measure 2, which is equivalent to the time span to be reconstructed. The trace length is varied from $120s$ to $360\,000s$ (approx. 4.2 days). The graph shows a major influence of the trace length on reconstruction accuracy. Very short traces of just $120s$ enable a very accurate reconstruction but hold little practical value. Accuracy drops drastically for trace length $360s$, due to an unexpected behavior shift in the
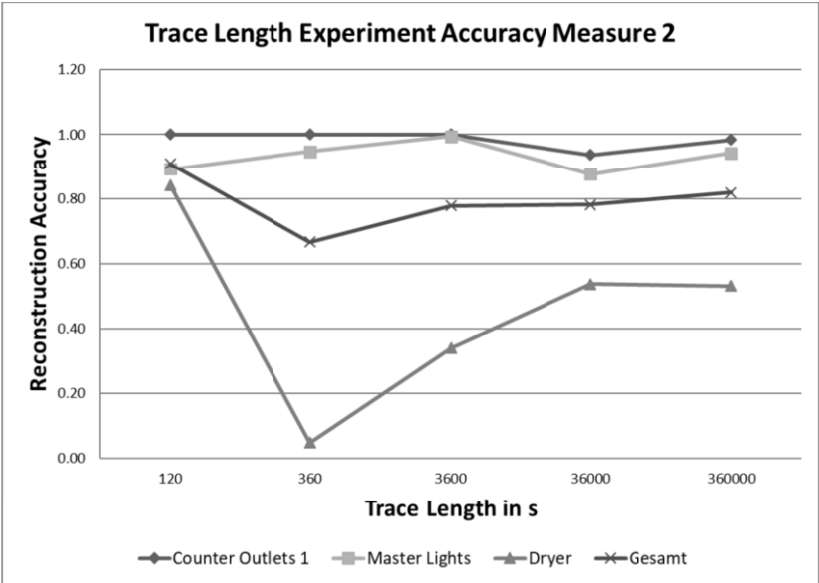
Fig. 5. Influence of Trace Length on Reconstruction Accuracy Overall and for Selected Appliances

Dryer on the chosen test day. Longer traces of $3\,600h$ to $36\,000h$ improve the accuracy, and hold practical value. A trace length of several days yields no further improvement in accuracy, and therefore trace lengths of $3\,600s$ and $36\,000s$ will be tested in the performance experiment.

4.2 Performance Experiments - Reconstruction Accuracy

For the performance experiment, three models with the reductions explained on Section 3.2 were trained, one for each month, using five consecutive days each. These were tested against $3\,600s$ traces from twelve single days, four from each month. (see Table 2) In order to cover effects of randomness and also user behavior varying over time, both training and test data encompassed weekend and weekdays, and test data was selected from within and without the training period.

Month	Training Data	Test Data			
May	6.5. - 10.5.	6.5.	9.5.	18.5.	20.5.
June	3.6. - 7.6.	3.6.	5.6.	15.6.	17.6.
July	1.7. - 5.7.	1.7.	5.7.	14.7.	15.7.

Table 2
Data Selected for Training and Testing Purposes

Figure 6 compares accuracy *measure 1* and *2*, showing the average accuracy over the different test days that was achieved for each of the appliances and for the whole. In general, accuracy *measure 1* seems more strict, since here a small amount of samples per appliance leads to large penalties, when some of these are
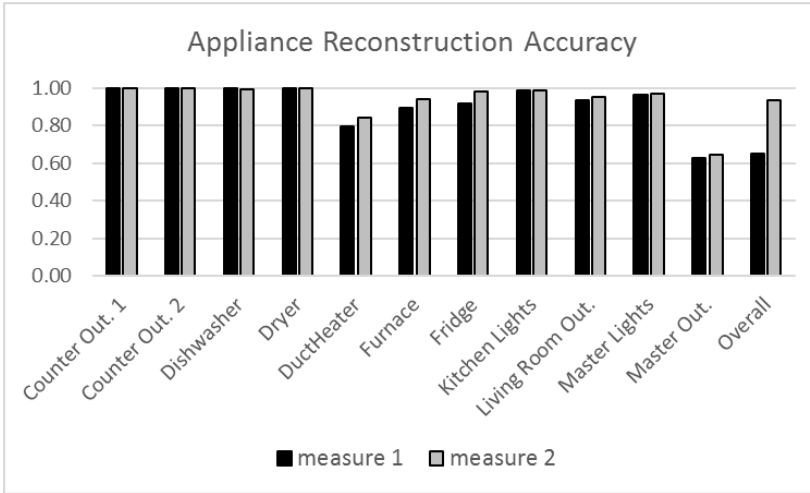
Fig. 6. Reconstruction Accuracy for Appliances and Overall for Accuracy Measure 1 and 2

reconstructed inaccurately. The overall accuracy of *measure 2* is much larger, which is due to more stable appliances, whose consumption is reconstructed accurately, balance the effect of other noisy and less accurately reconstructed consumers.

Most appliances could be reconstructed with an accuracy of 0.9 or better. The *Master Outlets* and the *Duct Heater* show the least accurate reconstructions. For the *Duct Heater* this was probably due to the extremely noisy data produced by the device, which make it inherently hard to reconstruct. A related reason is also that the noise could not be accurately represented in a three state model, and that a model with more appliance states for the Duct Heater could improve its reconstruction accuracy. This is however a topic for future research. The *Master Outlets* represent not a single appliance, but a variable set, where probably the actual appliances plugged in and used differed from training to test data.

The reconstruction accuracy for the whole model *measure 1* achieved per model and test day is shown in Figure 7. The first observation is that there seems to be little difference between the three models. The drastic decline of accuracy for mid June to mid July is due to the *Master Outlets* being reconstructed completely wrong. In all other cases the accuracy is 0.8 or above and thus acceptable.

The reconstruction accuracy for the whole model *measure 2* achieved per model and test day is shown in Figure 8. As now in every step every device state is compared, devices with a more stable behavior gain more influence, which also seem to be easier to reconstruct. As opposed to measure 1, here one can see a slight difference between the models, the July model performing better on the July test data than the June or May model. The decline of accuracy for mid June to mid July can also be observed here, but for the same reason as stated above it is not nearly as drastic.

The first set of performance experiments was performed with only $3\,600s$ of test data for each day. To try for a longer trace of $36\,000s$, the May model was applied to the reconstruction of two May test days. Each set took roughly 1 hour
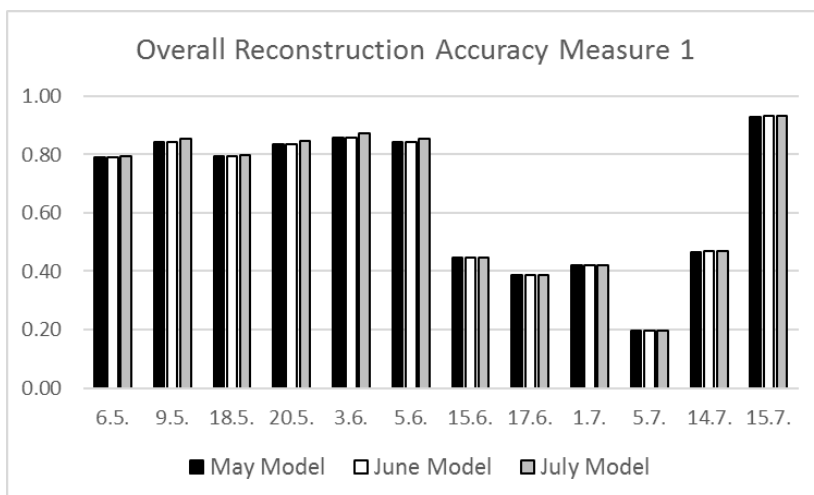
Fig. 7. Overall Reconstruction Accuracy Measure 1 for all Three Models and Twelve Test Days
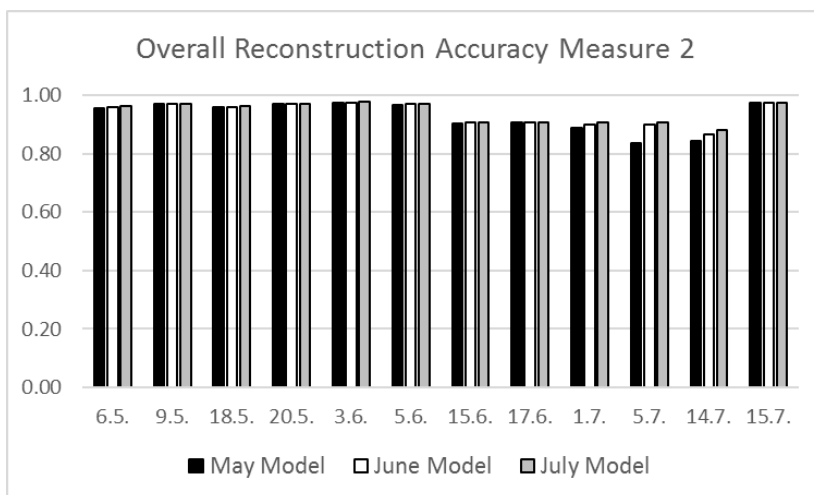

Fig. 8. Overall Reconstruction Accuracy Measure 2 for all Three Models and Twelve Test Days

of computation time. Figure 9 shows the accuracy *measure 2* achieved for the single appliances and overall compared between the $3\,600s$ and $36\,000$ test traces. The two test days seem comparable in accuracy, the reconstruction for May 18th from without the test set even being slightly better. Here the *Duct Heater* shows again that it is inherently hard to reconstruct due to very noisy data. Overall, the accuracy improves with increasing trace length, which is a positive result.

The experiments show that VSS can be applied to NIALM in the tested setting. The results are promising with an overall reconstruction accuracy of 0.94 for accuracy *measure 2*. This is especially notable since more than half of the appliances were abstracted and joined in a single unknown consumer.
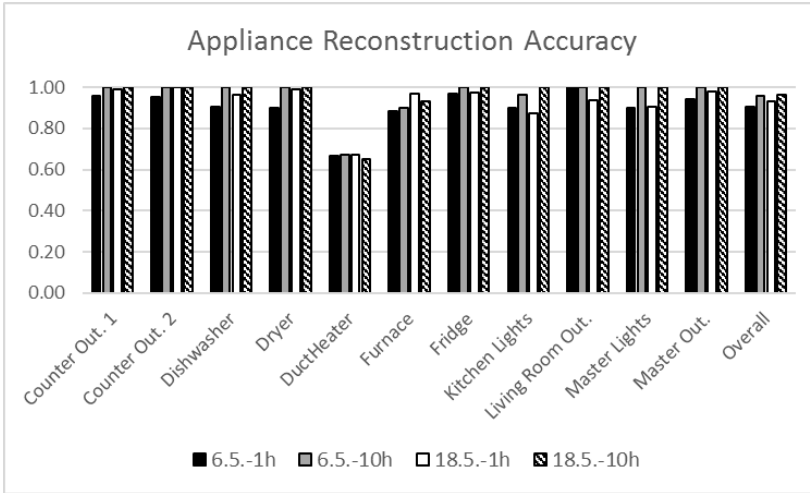
Fig. 9. Comparing May Model Appliances and Overall Reconstruction Accuracy for $3\,600s$ and $36\,000s$ Traces from May 6th and 18th

5   Conclusion

The paper introduced and tested the feasibility of applying Virtual Stochastic Sensor behavior reconstruction methods to non-intrusive appliance load monitoring. Models for states and energy consumption of the single appliances were constructed and then joined in a combined system model. Reducing the appliance set and the individual appliance state spaces was necessary to make the reconstruction feasible. These reductions are however not an inherent limitation of VSS, but became necessary due to run-time and memory consumption requirements of the Decoding task. The experiments determined suitable model parameters and showed that the behavior reconstruction can yield an accuracy of 0.9 or above for this setting. This is in the range of existing NIALM approaches and therefore, using VSS for NIALM can be considered feasible and competitive. [4]

5.1   Future Work

Building on the current paper, the approach needs to be tested on further data sets, e.g. the REDD data set should be examined in future experiments.

One major drawback of the approach shown here is that the reconstruction model needed to be trained / parametrized using the single appliances' consumption data. This is in the long term not practically feasible since it would require at least temporary installation of detailed sensory equipment, which NIALM is actually striving to make obsolete.

However, one idea to alleviate this problem is to start from a known (major) appliance setup in a household and use generalized or manufacturer appliance models for the reconstruction model, which could then be fine-tuned using the cumulative consumption data. This would reduce the necessary intrusion for model training or even eliminate it completely.

References

[1] Anderson, K., A. F. Ocneanu, D. Benitez, D. Carlson, A. Rowe and M. Berges, *Monitoring labeled public data for event-based non-intrusive load monitoring research*, in: *Proceedings of SustKDD'12, Beijing, China*, 2012.

[2] Baker, S., A. Mishra, D. Irwin, E. Cecchet, P. Shenoy and J. Albrecht, *Smart*: An open data set and tools for enabling reearch in sustainable homes*, in: *Proceedings of SustKDD'12, Beijing, China*, 2012.

[3] Baranski, M.-. and J. Voss, *Genetic algorithm for pattern detection in nialm systems*, in: *IEEE International Conference on Systems, Man and Cybernetics*, 2004.

[4] Batra, N., J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, A. Singh and M. Srivastava, *Nilmtk: An open source tootool for non-intrusive load monitoring*, in: *Proceedings of the fifth Internatonal COnference on Future Energy Systems, Cambridge, UK*, 2014.

[5] Buchholz, R., "Conversive Hidden Non-Markovian Models," Ph.D. thesis, submitted to Otto-von-Guericke-University Magdeburg (2012).

[6] Dittmar, T., C. Krull and G. Horton, *An improved conversive hidden non-markovian model-based touch gesture recognition system with automatic model creation*, in: *Proceedings of the 14th International Conference on Modeling and Applied Simulation, Bergeggi, Italy*, 2015.

[7] Dittmar, T., C. Krull and G. Horton, *A new approach for touch gesture recognition: Conversive hidden non-markovian models*, Journal of Computational Science (2015).

[8] Fink, G. A., "Markov Models for Pattern Recognition," Springer, Berlin, Heidelberg, 2008.

[9] Horton, G., *A new paradigm for the numerical simulation of stochastic petri nets with general firing times*, in: *Proceedings of the European Simulation Symposium 2002* (2002), pp. 129–136.

[10] Kang, S. J., K. Shin, R. Yang, S. Y. Lee and J. Sohn, *Cost effective disaggregation mechanisms for the nialm system*, International Journal of Smart Grid and Clean Energy (2013).

[11] Kim, H., M. Marwahy, M. Arlitty, G. Lyon and J. Han, *Unsupervised disaggregation of low frequency power measurements*, in: *Proceedings of the 11th SIAM International Conference on Data Mining,*, 2011.

[12] Kolter, J. Z. and M. J. Johnson, *Redd: A public data set for energy disaggregation research*, in: *Proceedings of SustKDD'11, San Diego, USA*, 2011.

[13] Kolter, Z. and T. Jaakkola, *Approximate inference in additive factor hmms with application to energy disaggregation*, in: *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS), La Palma, Canary Islands*, 2012.

[14] Krull, C., R. Buchholz and G. Horton, *Matching hidden non-markovian models: Diagnosing illnesses based on recorded symptoms*, in: *Proceedings of ESM 2010. 25th-27th Oktober 2010, Hasselt, Belgium*, 2010, pp. 133–138.

[15] Krull, C., R. Buchholz and G. Horton, *Virtual stochastic sensors: How to gain insight into partially observable discrete stochastic systems*, in: *The 30th IASTED International Conference on Modelling, Identification and Control, 14th-16th February 2011, Innsbruck. Austria*, 2011.

[16] Krull, C. and G. Horton, *Hidden non-markovian models: Formalization and solution approaches*, in: *Proceedings of 6th Vienna Conference on Mathematical Modelling, Vienna, Austria*, 2009, pp. 682–693.

[17] Krull, C., G. Horton, B. Denkena and B. Dengler, *Virtual stochastic sensors for reconstructing job shop production workflows*, in: K. Al-Begain, editor, *EUROSIM '13 Proceedings of the 2013 8th EUROSIM Congress on Modelling and Simulation* (2013), pp. 276–281.

[18] Lazarova-Molnar, S., "The Proxel-Based Method: Formalisation, Analysis and Applications," Ph.D. thesis, Otto-von-Guericke-University Magdeburg (2005).

[19] Nylund, J.-A. M., "Semi-Markov Modelling in a Gibbs Sampling Algorithm for NIALM," Master's thesis, Royal Institute of Technology, Stockholm, Sweden (2014).

[20] Parson, O., "Unsupervised Training for Non-intrusive Appliance Load Monitoring for Smart Meter Data," Ph.D. thesis, University of Southhampton, UK (2014).

[21] Parson, O., S. Ghosh, M. Weal and A. Rogers, *Non-intrusive monitoring using prior models if general appliance types*, in: *Proceedings of the Twenty-Sixth AAAI Conference in Artificial Intelligence*, 2012.

[22] Rabiner, L. R., *A tutorial on hidden markov models and selected applications in speech recognition*, in: *Proceedings of the IEEE, 2* **77**, 1989, pp. 257–286.

[23] Thiel, M., "Machbarkeitsstudie zur Anwendbarkeit von Hidden non-Markovian Modellen im Bereich NIALM," Masters thesis, Otto-von-Guericke-University Magdeburg (2015).

[24] Zeifman, M., C. Akers and K. Roth, *Nonintrusive appliance load monitoring (nialm) for energy control in residential buildings*, in: *Proceedings of International Conference on Energy Efficiency in Domestic Appliances and Lighting (EEDAL), Copenhagen*, 2011.

[25] Zeifman, M. and K. Roth, *Nonintrusive load monitoring: Review and outlook*, IEEE Transactions on Consumer Electronics (2011).