



D3.5 FINAL ENERGY DISAGGREGATION ALGORITHMS

Project title	Collaborative Recommendations and Adaptive Control for Personalised Energy Saving
Project acronym	enCOMPASS
Project call	EE-07-2016-2017 Behavioural change toward energy efficiency through ICT
Work Package	WP3
Lead Partner	SUPSI
Contributing Partner(s)	CERTH
Security classification	Public
Contractual delivery date	30/04/2019
Actual delivery date	29/04/2019
Version	1.0
Reviewers	CERTH, GRA

History of changes

Version	Date	Comments	Main Authors
0.1	26/01/2019	DDP (Deliverable Development Plan) – definition of the document structure and contributions expected from each partner.	A.E. Rizzoli, C. Rottondi
0.2	1/3/2019	Added Sections 1-3.	C. Rottondi
0.3	12/4/2019	Added Section 4.1, 4.2, 4.3, 4.4	C. Rottondi
0.4	16/4/2019	Document reformatiing and preparation for review	A.E. Rizzoli
0.5	17/4/2019	Added technical details to the section 3.3	M. Derboni
0.6	22/4/2019	Finalised the results of the synthetic simulation in Section 4.4.1	C. Rottondi, A.E. Rizzoli
0.7	25/4/2019	Added section 5 on the comfort inference	S. Krinidis
0.8	25/4/2019	Finalized version	A.E. Rizzoli & S. Krinidis
0.9	29/4/2019	Review	K. Tarjanyi
1.0	29/4/2019	Final Version	S. Krinidis

Disclaimer

This document contains confidential information in the form of the enCOMPASS project findings, work and products and its use is strictly regulated by the enCOMPASS Consortium Agreement and by Contract no. 723059.

Neither the enCOMPASS Consortium nor any of its officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

The contents of this document are the sole responsibility of the enCOMPASS consortium and can in no way be taken to reflect the views of the European Union.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723059.

TABLE OF CONTENTS

Executive Summary	5
1 Introduction.....	6
2 State of the art on energy end-use characterization	7
2.1 The Non Intrusive Appliance Load Monitoring (NIALM) Problem.....	7
2.2 Optimization-based NIALM approaches.....	9
2.3 Pattern recognition-based NIALM approaches	10
3 The enCompass algorithm for energy end-use disaggregation.....	11
3.1 Motivation	11
3.2 Quadratic Programming Model for energy disaggregation	11
3.2.1 Input sets	11
3.2.2 Input parameters.....	11
3.2.3 Output variables	12
3.2.4 Objective function	12
3.2.5 Problem constraints	12
3.3 Parameter training and QP model solution.....	13
4 Performance Assessment	15
4.1 Datasets	15
4.2 Scenarios and Benchmarks.....	15
4.3 Performance Metrics	16
4.4 Testing and Validation	16
4.4.1 Assessment with synthetic data	16
4.4.2 Assessment with UKDale dataset	37
4.4.3 Assessment on SES user data	51
5 Comfort Inference Engine	57
5.1 Thermal Comfort	57
5.1.1 Sensor Data preprocessing	61
5.1.2 User Feedback and prediction of Metabolic Rate and Activity	62
5.2 Visual comfort	68
5.2.1 Visual comfort Inference	70
5.2.2 Visual comfort Prediction	71
5.3 Results	74
6 Conclusions.....	80
7 Bibliography.....	81

EXECUTIVE SUMMARY

“D.3.5 Final energy disaggregation algorithms” is specified in the enCOMPASS Description of Action as defining “the initial prototype, with documentation, of the algorithms for energy disaggregation into end uses, updated after the validation in the pilots”.

This deliverable is one of the outputs of the project task “T3.4 Disaggregation of energy use”, which aims at developing algorithms to disaggregate the building's energy consumption to the level of individual users and appliances. This task is fed with the information provided by T 3.1 and T 3.2: it uses the aggregate energy consumption at building level at different temporal resolutions, and it also uses the so called “signatures” of specific appliances, to disaggregate the total consumption into its components. Accuracy will depend on the quality and resolution of the available input data, but the objective of this task is to study a set of algorithms able to gracefully degrade their performance, to provide useful results under a wide spectrum of situations. The output of the algorithms allows for adaptive, in-situ feedback on energy consumption and recommendations for energy saving actions. Furthermore, this deliverable also refers the comfort inference engine developed within encompass project (T3.3), since the output of this Task are not documented anywhere, and the consortium has decided to be delivered within this document.

For an overall description of the dependencies among the above task T3.4 and the other Project tasks and work packages, please refer to the section 3.1.2 “detailed work description” of the enCOMPASS proposal.

This deliverable presents the final version of the algorithms developed in the enCOMPASS project to derive, directly from metered energy consumption data, mathematical models describing the users' consumption behavior. Specifically, the algorithms focus on energy end-use characterization, which aims at decomposing the aggregate (i.e., whole household) high-resolution energy flow data collected from a single measurement point into energy end use categories (e.g., washing machine, dishwasher), in order to understand how, when and where energy is used. The developed disaggregation algorithms are tested against data available in the literature or synthetically generated by open source software emulators of residential energy consumption traces.

This document is structured as follows:

- Section 1 describes the goals and motivations of end-use energy disaggregation approaches.
- Section 2 provides a review on the state-of-the-art algorithms for energy end-use disaggregation.
- Section 3 describes the final version of the algorithms for energy end-use characterization developed within the enCOMPASS project.
- Section 4 discusses the performance assessment of the algorithm by testing it on real energy consumption data.
- Section 5 describes the comfort inference engine developed within encompass project.
- Section 6 concludes the document.

1 INTRODUCTION

The introduction of smart meters makes possible to collect energy consumption readings at fine-grained spatio-temporal resolution (i.e., measurements with granularity in the order even of a few seconds, for single households), thus enabling the extraction of detailed information about individual energy usage habits. In turn, such knowledge allows for the construction of more accurate mathematical models to characterize individual and collective energy consumption behaviors.

In the framework of the enCOMPASS project, Work Package 3 addresses energy end-use disaggregation, which aims at breaking down the total energy consumption measured at household level into the contributions of single electrical appliances. The use of such disaggregated information is twofold: on one side, it can be used to develop predictive models capable of forecasting future energy consumption behaviours, on the other side it can be directly provided to customers, so that household's components gain a detailed knowledge of their energy usage. For instance, through the enCOMPASS platform and app, customers can visualize their hourly consumption, as well as charts on their energy end-uses patterns across major end-use categories (e.g., washing machine, dishwasher, clothes dryer, fridge) and they can be alerted of occurring consumption anomalies. Furthermore, personalized hints for reducing energy consumption can be directly delivered by means of the recommender system, which is described in Deliverable D4.2 First user behavior model and recommender. These stimuli are aimed at fostering the adoption of energy saving actions, such as replacing low-efficient appliances into high-efficient ones and reducing energy waste (e.g. turning off lights when rooms are empty).

2 STATE OF THE ART ON ENERGY END-USE CHARACTERIZATION

2.1 THE NON INTRUSIVE APPLIANCE LOAD MONITORING (NIALM) PROBLEM

There is a rich literature on automatic disaggregation methods (known as Non-Intrusive Appliance Load Monitoring – NIALM – algorithms) aimed at decomposing the aggregate household energy consumption data collected from a single measurement point into device-level consumption data (see Figure 1 for a simple example), requiring limited or even no interaction with the user.

According to (Zeifman & Roth, 2011), electric appliances can be categorized into four distinct groups, depending on the characteristics of their energy consumption patterns:

- *Always-on appliances* exhibit constant and continuous energy consumption patterns. Appliances such as smoke detectors and telephone sets belong to this category.
- *On/off appliances* have only two operational states: their consumption is null when they are not in use and exhibit constant consumption when they are turned on. Therefore, their consumption patterns along time have rectangular shapes. This category includes appliances such as table lamps (without dimming capabilities), toasters, electric kettles etc.
- *Multistate appliances* exhibit a finite number of consumption levels, when they are in on state. However, the transition between states often shows repeatable patterns, whose knowledge can be exploited to ease the disaggregation process. Examples of appliances belonging to this category include dishwashers, washing machines, heaters, fridges etc.
- *Continuously variable appliances* may change their energy consumption level among an infinite (or very high) number of states and show little or no repeatability in their consumption patterns. This category includes e.g. electric drills and lights with dimming capabilities.

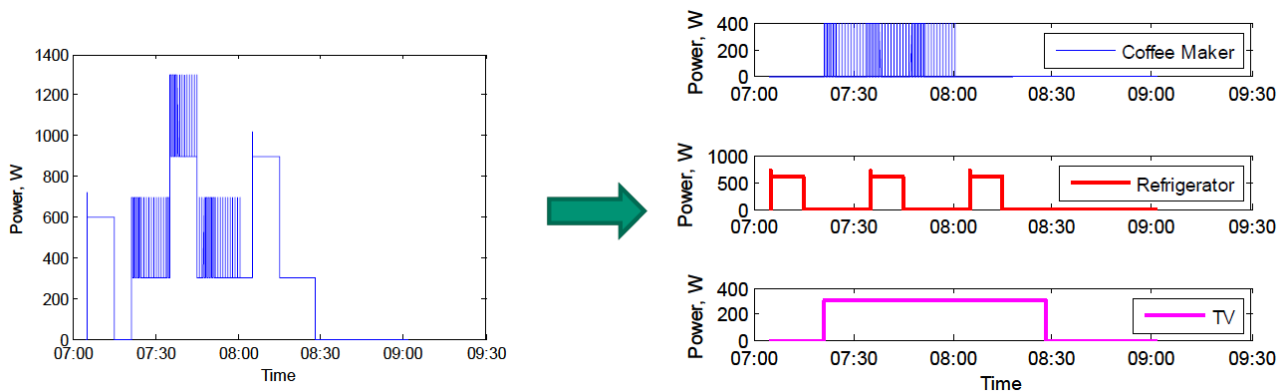


Figure 1: Example of inputs (left) and outputs (right) of a NIALM algorithm (Zeifman & Roth, 2011)

The first algorithm for NIALM was proposed by Hart in 1992 (Hart, 1992). Hart’s approach is based on the segmentation of the aggregate power signal into successive steps, which are then matched to the appliance signatures. However, this method is not able to detect multistate appliances and it is neither able to decompose power signals made of simultaneous on/off events on multiple appliances. Since Hart’s contribution, the NIALM problem has been extensively studied in the literature. The survey papers (Zoha, Gluhak, Imran, & Rajasegarar, 2012) and (Zeifman & Roth, 2011) give a complete review on the state-of-the-art of NIALM methods.

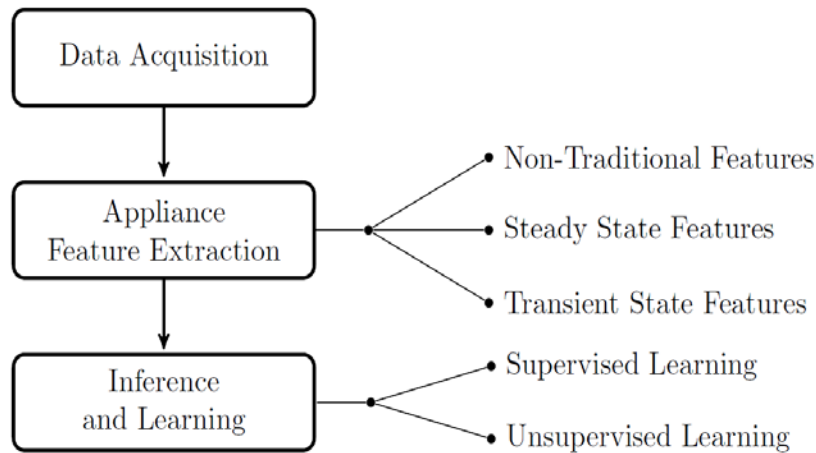


Figure 2: Decomposition of the NIALM problem into tasks (Zoha, Gluhak, Imran, & Rajasegarar, 2012)

Broadly speaking, the NIALM problem can be decomposed in three main tasks, as showed in Figure 2:

- *data acquisition*: this task consists in gathering aggregated consumption measurements of a household via properly configured smart meters. Depending on the smart meter hardware and computational capabilities, various types of data with different sampling frequencies can be collected. Data such as real power, reactive power, Root Mean Square (RMS) voltage and current values can be computed at sampling frequencies above 100 Hz. Conversely, transient events and power noise can only be detected if the sampling rate of the meter is above a few MHz (which requires more expensive hardware configurations w.r.t. standard commercially available smart meters). Once the measurements are collected, they can either be processed locally or transmitted to an external processing device through a telecommunication network via a network interface card. Other data that could be used as additional inputs to the disaggregation process are in-home sensor measurements such as temperature, luminance, motion or even partially disaggregated energy consumption readings measured by smart plugs. Again, frequency and resolution of such data largely depend on the sensor type and computational resources.
- *appliance feature extraction*: after collection, raw measurements must be processed to extract e.g. power-related metrics such as active and reactive power consumption, steady-state or transient state transitions. Similarly, sensor data can be elaborated to extract e.g. presence/absence patterns.
- *inference and learning*: once appliance features have been extracted, they are analysed by a load identification algorithm in order to perform disaggregation. Disaggregation algorithms apply either supervised or unsupervised learning methods. As of today, the majority of supervised methods apply either *optimization* based or *pattern recognition* based approaches. Optimization based approaches try to identify candidate combinations of individual consumption patterns such that their sum shows a trend similar to the observed aggregated measurements. The main drawback is that the presence of unknown loads may easily degrade the performance of such algorithms. Therefore, pattern recognition approaches have been developed, which try to match the extracted features with a pool of known or pre-acquired load signatures. Note that both approaches require a proper amount of training data to allow for an initial training phase. Such requirement is often unlikely to be fulfilled in real settings, as load signatures typically show significant variation depending on the specific electrical device, so that a one-size-fit-all training rarely achieves high disaggregation accuracy. To overcome this issue, unsupervised methods for load disaggregation have been investigated, which attempt to

breakdown the aggregated consumption measurements into single appliances' contributions, without performing any sort of event detection.

In the following Subsection, a brief overview of the above mentioned disaggregation approaches will be provided. Note that the vast majority of the studies on NIALM algorithms validate the proposed solutions using publicly accessible datasets of real energy consumption measurements. The most widely used datasets made available in the last years are reported in Table 1. Alternatively, synthetic load consumption traces generated by open source software such as Loadprofilegenerator¹ can be adopted.

Table 1: List of publicly available datasets of disaggregated energy consumption measurements.

Authors	Title	Venue	Year
J. Z. Kolter and M. J. Johnson	REDD: A Public Data Set for Energy Disaggregation Research	1 st KDD Workshop on Data Mining Applications in Sustainability	2011
K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges	BLUED: a fully labeled public dataset for Event-Based Non-Intrusive load monitoring research	2 nd KDD Workshop on Data Mining Applications in Sustainability	2012
S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy and J. Albrecht	Smart*: An open data set and tools for enabling research in sustainable homes	2 nd KDD Workshop on Data Mining Applications in Sustainability	2012
S. Makonin, F. Popowich, L. Bartram, B. Gill, B. and I. Bajic	AMPds: a public dataset for load disaggregation and eco-feedback research	IEEE Electrical Power and Energy Conference	2013
A. Monacchi, D. Egarter, W. Elmenreich, S. D'Alessandro, S. and A. M. Tonello	GREEND: An energy consumption dataset of households in Italy and Austria	IEEE International Conference on Smart Grid Communications	2014
J. Kelly and W. Knottenbelt	The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes	Scientific data, Volume 2, article number 150007	2015

2.2 OPTIMIZATION-BASED NIALM APPROACHES

In optimization-based methods the disaggregation problem is formulated as an optimization model. In its basic flavour (i.e., the recognition of a single working appliance), the problem consists in comparing the extracted feature vector \vec{x} to those of known loads \vec{y}_i pre-stored in a database and choosing the most similar one based on the minimization of some kind of error metric (typically the squared error). Note that the database may contain multiple feature vectors for every appliance type, as its energy consumption curves may

¹ Free download available at www.loadprofilegenerator.de (accessed on March 31, 2019)

differ depending on the device manufacturer and on the specific usage (e.g., the duration and type of washing cycle of a dishwasher). Therefore, the objective function can be formulated as: $\arg \min_i \|\vec{x} - \vec{y}_i\|$.

In presence of several working appliances, the complexity of the optimization problem increases, as combination of multiple candidate appliances included in the database must be taken into account. In such scenarios, ensuring scalability in terms of computational complexity and time becomes challenging, especially in presence of unknown loads. Recent studies considered different optimization approaches including integer programming (Suzuki, Inagaki, Suzuki, Nakamura, & Ito, 2008), (Camier, Giroux, Bouchard, & Bouzouane, 2013), sparse coding (Figueiredo, Ribeiro, & de Almeida, 2013), (Dong, Ratliff, Ohlsson, & Sastry, 2013) and genetic algorithms (Baranski & Voss, 2004).

2.3 PATTERN RECOGNITION-BASED NIALM APPROACHES

Pattern matching approaches still leverage a database containing multiple appliance specific features that used as inputs by the disaggregation algorithm. The simplest ones rely on clustering methods such as the one proposed in Hart's seminal paper (Hart, 1992), which groups appliances in clusters in a 2D plane based on their active-reactive power consumption. An unknown load is identifiable by mapping it onto a point of the 2D space based on its steady-state consumption characteristics, calculating its distance from the centroid of every cluster according to a predefined metric and choosing the nearest one. Thanks to their simplicity, clustering methods have been widely investigated in the scientific literature related to NIALM. To address their main drawback, i.e. the impossibility of recognizing unknown loads, alternative approaches based on Bayesian inference, Hidden Markov Models, Artificial Neural Networks and Support Vector Machines have been proposed (see e.g. (Marchiori, Hakkarinen, Han, & Earle, 2011), (Srinivasan, Ng, & Liew, 2006), (Zia, Bruckner, & Zaidi, 2011), (Parson, Ghosh, Weal, & Rogers, 2012), (Johnson & Willsky, 2013)).

3 THE ENCOMPASS ALGORITHM FOR ENERGY END-USE DISAGGREGATION

This Section provides a description of the final version of the disaggregation algorithm that has been integrated in the enCOMPASS platform to decompose the aggregated energy consumption readings of a household in the consumption patterns of individual appliances. The Section is organized as follows: the energy disaggregation problem is formalized in Subsection 3.2, whereas the training procedure adopted to tune the problem parameters and the procedure used to solve the disaggregation problem are discussed in Section 3.3.

3.1 MOTIVATION

The initial version of the load disaggregation algorithm presented and validated in deliverable D3.3 – First energy disaggregation algorithm was based on the approach described in (Piga, Cominola, Giuliani, Castelletti, & Rizzoli, 2016), which exploited the assumption that the power demand profiles of each appliance are piecewise constant over time. The disaggregation problem was treated as a least-square error minimization problem, with an additional (convex) penalty term aiming at enforcing the disaggregated signals to be piecewise constant over time. However, the assumption of piece-wise constant pattern behaviour is less likely to hold when considering the energy measurement granularity adopted in the enCOMPASS smart metering system (i.e., 15 min resolution). Moreover, the approach by Piga et al. cannot be applied in presence of unknown loads. Unfortunately, though in the enCOMPASS pilots a list of the main appliances owned by each user was acquired by means of a questionnaire that the users filled in at the beginning of the intervention period, knowledge of the full electrical equipment available in each household participating to the pilots could not be gathered. Therefore, the initial version of the load disaggregation algorithm was evolved to take into account the presence of unknown electrical devices. In the following, we formalize the final version of the energy end-use disaggregation problem as a quadratic programming (QP) model.

3.2 QUADRATIC PROGRAMMING MODEL FOR ENERGY DISAGGREGATION

We now define the problem inputs (sets and parameters), the output variables, the objective function and the problem constraints.

3.2.1 Input sets

- T = sets of time epochs ($t=1,2,\dots,|T|$)
- A = set of appliances
- L_a = sets of energy consumption levels of appliance $a \in A$

3.2.2 Input parameters

- c_t = aggregate energy consumption during time epoch $t \in T$
- m_a =maximum daily energy consumption of appliance $a \in A$
- d_a =maximum daily usage duration (i.e., maximum number of consecutive epochs in the appliance is on) of appliance $a \in A$
- w_a =minimum daily usage duration (i.e., minimum number of consecutive epochs in which the appliance is on) of appliance $a \in A$
- $u_{a,t}$ = binary parameter, set to 1 if appliance $a \in A$ can be turned on at time $t \in T$
- α_a = multiplicative weight of appliance $a \in A$

3.2.3 Output variables

- $x_{a,l,t}$ = binary variable, it is set to 1 if appliance $a \in A$ operates at consumption level $l \in L_a$ during time epoch $t \in T$
- $y_{a,t}$ = binary variable, it is set to 1 if appliance $a \in A$ changes consumption level at time epoch $t \in T$
- $o_{a,t}$ = binary variable, it is set to 1 if appliance $a \in A$ is on at time epoch $t \in T$
- f_a = binary variable, it is set to 1 if appliance $a \in A$ is on during at least one time epoch during the considered time horizon
- wm = integer variable, indicates the last epoch of activity of the washing machine
- cd =integer variable, indicates the first epoch of activity of the clothes dryer

3.2.4 Objective function

$$\min \sum_{t \in T} \left(c_t - \sum_{a \in A, l \in L_a} l \cdot u_{a,t} \cdot x_{a,l,t} \right)^2 + \sum_{t \in T, a \in A} \alpha_a \cdot y_{a,t}$$

The objective function minimizes the sum of two contributions: the first one is the quadratic error (i.e., the difference between the observed aggregated measurement and the sum of the reconstructed consumption of every appliance, at every time epoch), the second one is a penalty for every change of consumption level experienced by each appliance during the optimization horizon. By tuning the weights α_a , the penalty attributed to a non-piecewise-constant energy consumption of certain appliances can be strengthened or relaxed.

3.2.5 Problem constraints

- $\sum_{l \in L_a} x_{a,l,t} = 1 \quad \forall t \in T, a \in A$

This constraint imposes that each appliance operates at a single energy consumption level during each time epoch.

- $y_{a,t} \geq u_{a,t} \cdot x_{a,l,t} - u_{a,t} \cdot x_{a,l,t-1} \quad \forall t \in T, a \in A, l \in L_a: t > 1$
- $y_{a,t} \geq u_{a,t} \cdot x_{a,l,t-1} - u_{a,t} \cdot x_{a,l,t} \quad \forall t \in T, a \in A, l \in L_a: t > 1$

These constraints set variable $y_{a,t}$ to 1 if appliance $a \in A$ changes consumption level at epoch $t \in T$.

- $\sum_{l \in L_a, t \in T} l \cdot u_{a,t} \cdot x_{a,l,t} \leq m_a \quad \forall a \in A$

This constraint imposes that the daily energy consumption of appliance $a \in A$ does not exceed the daily limit.

- $\sum_{l \in L_a} l \cdot u_{a,t} \cdot x_{a,l,t} \leq \max_{l \in L_a} l \cdot o_{a,t} \quad \forall a \in A, t \in T$

This constraint sets variable $o_{a,t}$ to 1 if appliance $a \in A$ is on at epoch $t \in T$.

- $o_{a,t} \cdot t - o_{a,t-1} \cdot (t-1) \leq d_a [1 - |T| \cdot (o_{a,t} + o_{a,t-1} - 2)] \quad \forall a \in A, t \in T, t' \in T: t > t'$

This constraint imposes that the maximum usage duration of appliance $a \in A$ does not exceed d_a .

- $f_a \cdot |T| \geq \sum_{l \in L_a, t \in T} l \cdot u_{a,t} \cdot x_{a,l,t} \quad \forall a \in A$

This constraint ensures coherence between the values of variable $x_{a,l,t}$ and of variable f_a .

- $\sum_{t \in T} o_{a,t} \geq w_a \cdot f_a \quad \forall a \in A$

This constraint imposes that the daily energy consumption of appliance $a \in A$ (if activated) is not lower than the daily lower limit w_a . This way, the disaggregation of load curves of appliances such as dishwasher, washing machine and clothes dryer takes into account the minimum duration of a washing/drying cycle.

- $wm \geq o_{wm,t} \cdot t \quad \forall t \in T$

This constraint sets variable wm to the last epoch of activity of the washing machine (if the washing machine is activated during the day).

- $cd \leq o_{cd,t} \cdot t + |T| \cdot (1 - o_{cd,t}) \quad \forall t \in T$

This constraint sets variable cd to the first epoch of activity of the clothes dryer (if the cloth dryer is activated during the day).

- $cd \geq wm + 1$

This constraint imposes that the clothes dryer is turned on after the end of the operational period of the washing machine.

- $\sum_{t \in T} u_{a,t} \cdot x_{a,l,t} \geq f_a \quad \forall a \in \tilde{A}, l' = \max_{l \in L_a} l$

This constraint imposes that each appliance belonging to set \tilde{A} works at the highest energy consumption level for at least one time epoch, if activated during the day. In our formulation, set \tilde{A} contains dishwasher, washing machine and clothes dryer. The energy consumption profiles of a typical operation cycle of such appliances normally include one or multiple peak consumption periods, corresponding e.g. to water heating or spinning. Therefore, this constraint imposes that at least one peak consumption epoch is included in the disaggregated consumption profile of such appliances.

- $\sum_{a \in A, l \in L_a, t \in T} l \cdot u_{a,t} \cdot x_{a,l,t} \leq \sum_{t \in T} c_t$

This constraint imposes that the sum of the disaggregated energy consumption profiles do not exceed the total energy usage measured by the smart meter located at the user's premises.

3.3 PARAMETER TRAINING AND QP MODEL SOLUTION

We now discuss how each input set and parameter of the QP model discussed in Section 3.2 is dimensioned.

- Set T : the number of epochs depends on the duration of the scheduling horizon and on the resolution of the aggregated consumption measurements collected by the smart meters. Assuming that the scheduling horizon is 24 hours and the granularity of consumption measurements is 15 mins, the number of epochs is 96, therefore we can define set $T = \{1, 2, \dots, 96\}$.
- Set A : the set of main electric appliances owned by each customer has been declared in an initial questionnaire at the beginning of the intervention period. Those appliances may include: dishwasher, washing machine, clothes dryer, oven, electric vehicle, heat pump, air conditioner.
- Set L_a : we assume that each appliance can operate at a predefined number of consumption levels. The number of levels and the energy consumption per epoch associated to each level can be determined by collecting statistics over historical individual consumption data (if available) or over publicly available datasets containing load consumption curves of the main categories of electrical appliances (see Section 4.1). Note that set L_a always contains the element 0 (corresponding to the appliance off state). In the following, we report the algorithm we used to extract consumption levels from consumption curves of individual appliances, when available to be used as training data.

1. Create an histogram by defining a set of energy consumption bins and computing the number of measurements falling into each bin, where the number of bins is a predefined system parameter (e.g., 50 bins of width 100 kW in the range 0-5000 kW);
 2. Identify the histogram peaks with prominence greater than p measurements, where p is a predefined system parameter and depends on the total number of available measurements, i.e. on the temporal window covered by the training dataset;
 3. Retrieve the extremes $[b_{low}, b_{high}]$ of the energy consumption bins associated to the selected peaks, calculate the corresponding energy consumption level as $(b_{high} - b_{low})/2$.
- Parameter c_t : aggregate energy consumption measurements are collected by smart meters installed at the users' premises. Note that, in case disaggregated consumption measurements collected via smart plugs are available, those are subtracted from c_t and disaggregation is performed excluding the directly monitored appliances from set A .
 - Parameters m_a , d_a and w_a : as for set L_a , the maximum daily energy consumption and minimum/maximum duration of the operational period of each appliance can be calculated either based on historical individual consumption patterns or on publicly available datasets. In our implementation, maximum and minimum durations were computed by identifying the epochs of activity of every appliance within the training dataset, computing the minimum (resp. maximum) number of consecutive activity epochs in the dataset and setting the values of w_a and d_a accordingly. To set the value of m_a , the average energy consumption c_{aver} during the activity epochs was calculated and we set $m_a = c_{aver} \times d_a$.
 - Parameter $u_{a,t}$: this parameter can be used to prevent some appliances from being turned on at certain time periods. For example, if absence from home is inferred by motion detectors, the off state of oven, dishwasher, washing machine and clothes dryer (unless they support automatic deferral of their operational period) can be enforced.
 - Parameter α_a : the value of the coefficients used to impose piecewise linear behaviour of the consumption curve was tuned depending on the appliance type. For example, the recharge of an electric vehicle does not typically show abrupt variations during the charging period (especially if the charger does not support multiple charging rates), whereas washing machines and dishwashers exhibit more pronounced energy consumption fluctuations, depending on the phase of the washing cycle (e.g. water heating, spinning, etc.). Moreover, as the main objective is the minimization of the quadratic error, weights were chosen so that the term $\sum_{t \in T, a \in A} \alpha_a \cdot y_{a,t}$ was at least one order of magnitude lower than the term $\sum_{t \in T} (c_t - \sum_{a \in A, l \in L_a} l \cdot x_{a,l,t})^2$ (i.e., if multiple solution minimizing the objective function exist, the one ensuring minimum value of $\sum_{t \in T, a \in A} \alpha_a \cdot y_{a,t}$ is selected).

The QP optimization model has been implemented in AMPL and solved using the Gurobi solver, running on a Linux machine with 2 x Intel Xeon E5-2620 v4 2.1GHz (20/32 cores have been allocated) and 16 GB of RAM. In order to guarantee a timely interaction with the modules of the enCOMPASS platform leveraging the outputs of the disaggregation engine (i.e., the recommender system and the user tracking algorithms), a maximum limit of 90 seconds to the computational time allowed for solving one instance was imposed (note that one problem instance per user must be solved on daily basis and the number of users included in the treatment groups of the three pilot deployments is around 300, which imposes practical limits on the overall computational time for the disaggregation process).

4 PERFORMANCE ASSESSMENT

4.1 DATASETS

The disaggregation algorithm presented in Section 3.2 has been validated using two different datasets:

- A set of synthetic traces generated using the publicly available `loadprofilegenerator`² software. The software includes 65 predefined generation models of household energy consumption traces, depending on the number, age and working habits of the dwellers. The geographic location and weather conditions are customizable, e.g. by retrieving historical weather data based on the selected calendar dates. For the sake of validation, we generated two months of consumption traces for each of the 65 models, in the period between July 1, 2015 and August 31, 2015.
- The UKDale 2015 dataset (see reference reported in Table 1), containing consumption measurements of 6 houses for different time periods. Three out of those (building 3, 4 and 6) were monitored for a period shorter than two months, thus we excluded them from our analysis. For the remaining 3 buildings, we considered the following periods: building 1 from April 1, 2013 to May 31, 2013, building 2 from May 1, 2013 to June 30, 2013, building 5 from July 1, 2014 to August 31, 2014.

Moreover, the proposed disaggregation algorithm has been implemented in the enCOMPASS framework and used to disaggregate metering data collected from the households involved in the three pilot deployments. In this Section, we report results obtained on a pool of 80 users participating to the Swiss pilot. The disaggregation was performed using measurements at 15 mins granularity, during the period between October 1, 2018 and January 31, 2019, and considering 8 different categories of appliances: fridge, washing machine, dishwasher, clothes dryer, electric car, heat pump, air conditioner.

4.2 SCENARIOS AND BENCHMARKS

In the numerical assessment, we consider two different scenarios when disaggregating the synthetic and UKDale dataset:

1. disaggregation of the 5 top consuming appliances, which are identified beforehand based on the individual consumptions during the training period;
2. disaggregation of a fixed set of appliances including: fridge, dishwasher, electric oven, air conditioner, washer-dryer.

The latter scenario replicates the working conditions of the disaggregation algorithm within the enCOMPASS framework, whereas in the former scenario the quality of disaggregation is expected to be higher, as the energy consumption component due to appliances not included in the 5 top consuming ones is likely to be limited, thus easing the disaggregation task.

The performance of the disaggregation algorithm described in Section 3.2, referred to as ILP in the following, is compared to that obtained by two state of the art disaggregation approaches implemented in the publicly available NILMTK framework³ (Batra, et al., 2014). The first one is based on Combinatorial Optimization (CO),

² Software freely downloadable at: www.loadprofilegenerator.de

³ Open source software freely downloadable at: <http://nilmtk.github.io/>

the second one on a Factorial Hidden Markov Model (FHMM) (see (Batra, et al., 2014) for further details on their implementation).

The CO and FHMM models are implemented in python, whereas the ILP model has been implemented in AMPL⁴ and solved with the Gurobi⁵ solver. A computational time limit of 180 s per instance was imposed.

4.3 PERFORMANCE METRICS

The following performance metrics, proposed in (Batra, et al., 2014), have been used to compare the performance of the three disaggregation algorithms:

The *Fraction of Total Energy Assigned Correctly (FTEAC)*, defined as:

$$FTEAC = \sum_{i=1, \dots, N} \min \left(\frac{\sum_{t=1}^T \hat{y}_i(t)}{\sum_{i=1}^N \sum_{t=1}^T \hat{y}_i(t)}, \frac{\sum_{t=1}^T y_i(t)}{\sum_{i=1}^N \sum_{t=1}^T y_i(t)} \right)$$

The *Normalized Error in Assigned Energy (NEAE)* for each appliance i , defined as:

$$NEAE_i = \frac{\sum_{t=1}^T |y_i(t) - \hat{y}_i(t)|}{\sum_{t=1}^T y_i(t)}$$

The *Root Mean Square Error (RMSE)* for each appliance i , defined as:

$$RMSE_i = \sqrt{\frac{1}{T} \sum_{t=1}^T (y_i(t) - \hat{y}_i(t))^2}$$

The *True/False Positive Rate (TPR/FPR)* for each appliance i , defined as:

$$TPR_i = \frac{TP}{TP + FN}; \quad FPR_i = \frac{FP}{FP + TN}$$

Where: $TP_i = \sum_{t=1}^T \text{AND}(y_i(t) = on, \hat{y}_i(t) = on)$; $TN_i = \sum_{t=1}^T \text{AND}(y_i(t) = off, \hat{y}_i(t) = off)$; $FP_i = \sum_{t=1}^T \text{AND}(y_i(t) = on, \hat{y}_i(t) = off)$; $FN_i = \sum_{t=1}^T \text{AND}(y_i(t) = off, \hat{y}_i(t) = on)$.

The *Accuracy (ACC)* and *Precision (PRE)* for for each appliance i , defined as:

$$ACC_i = \frac{TP + TN}{TP + TN + FP + FN}; \quad PRE_i = \frac{TP}{TP + FP}$$

4.4 TESTING AND VALIDATION

4.4.1 Assessment with synthetic data

We start comparing the performance of the three algorithms measured by means of the metrics listed in Section 4.3, considering the 5 top consuming appliances per building and different granularities of the input measurements (i.e., 5, 15, 30, 45 and 60 mins). As the appliances belonging to the top consuming set differ

⁴ <https://ampl.com/>

⁵ <http://www.gurobi.com/>

from building to building, global metrics are computed by averaging the results obtained for appliances belonging to each building.

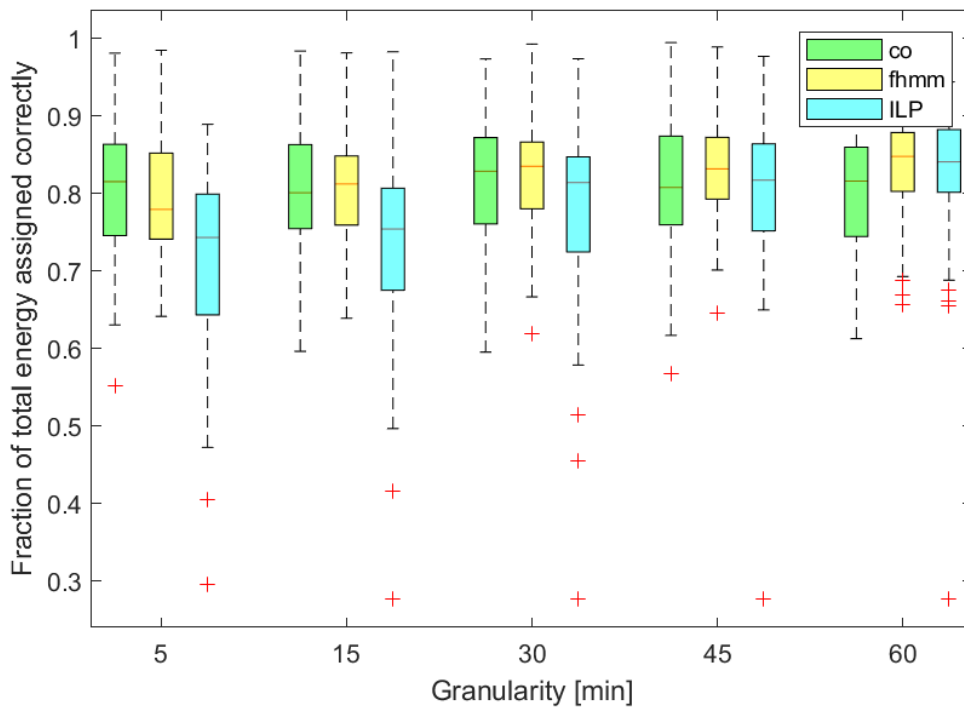


Figure 3: Fraction of Total Energy Assigned Correctly by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

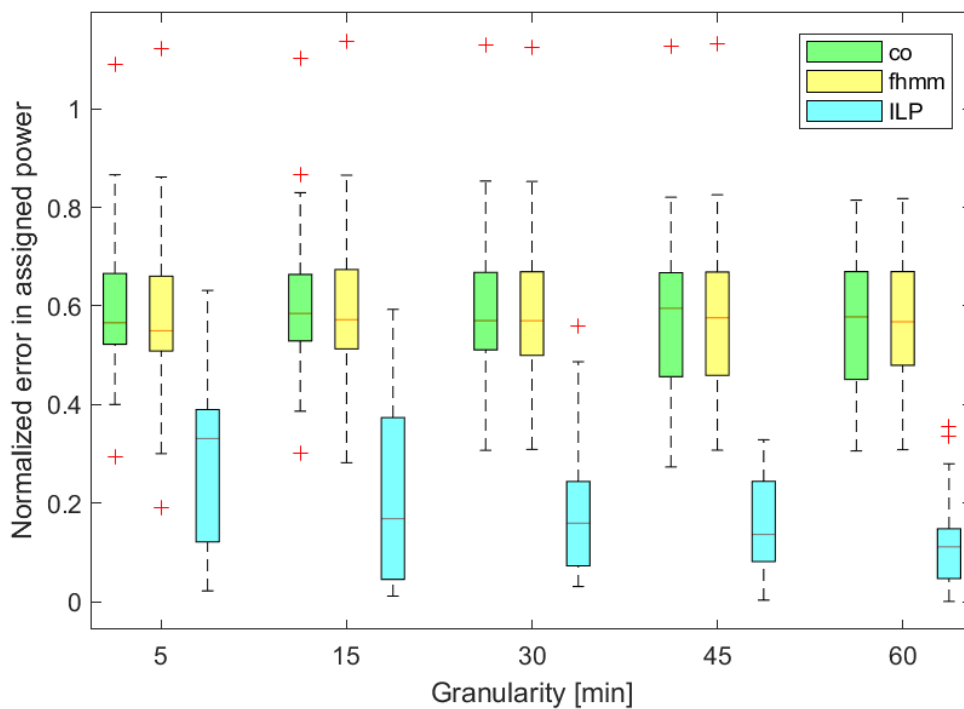


Figure 4: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

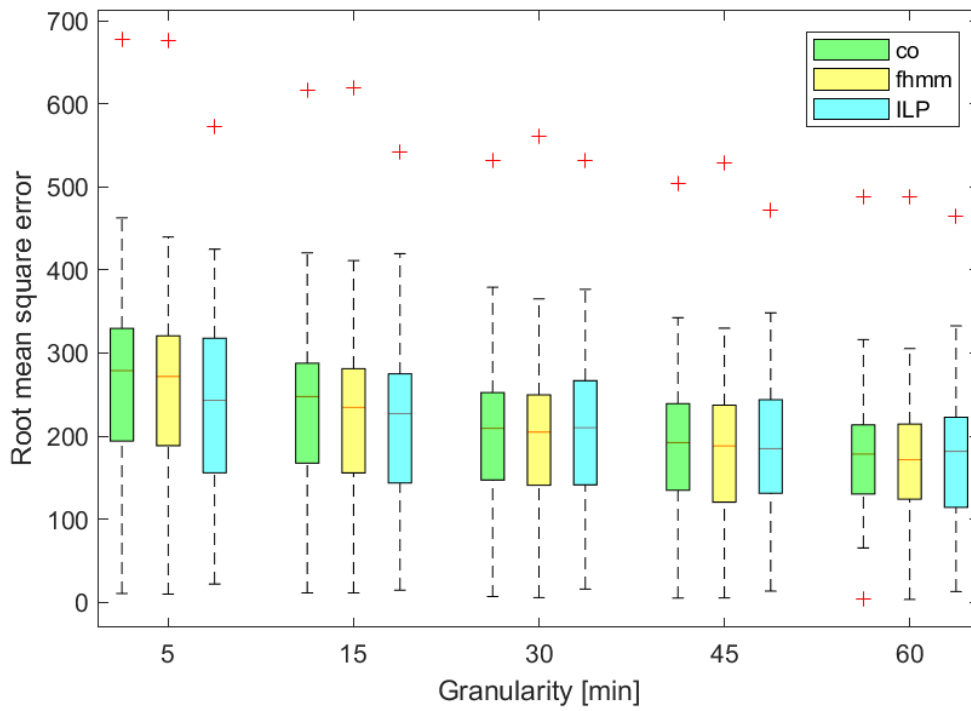


Figure 5: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

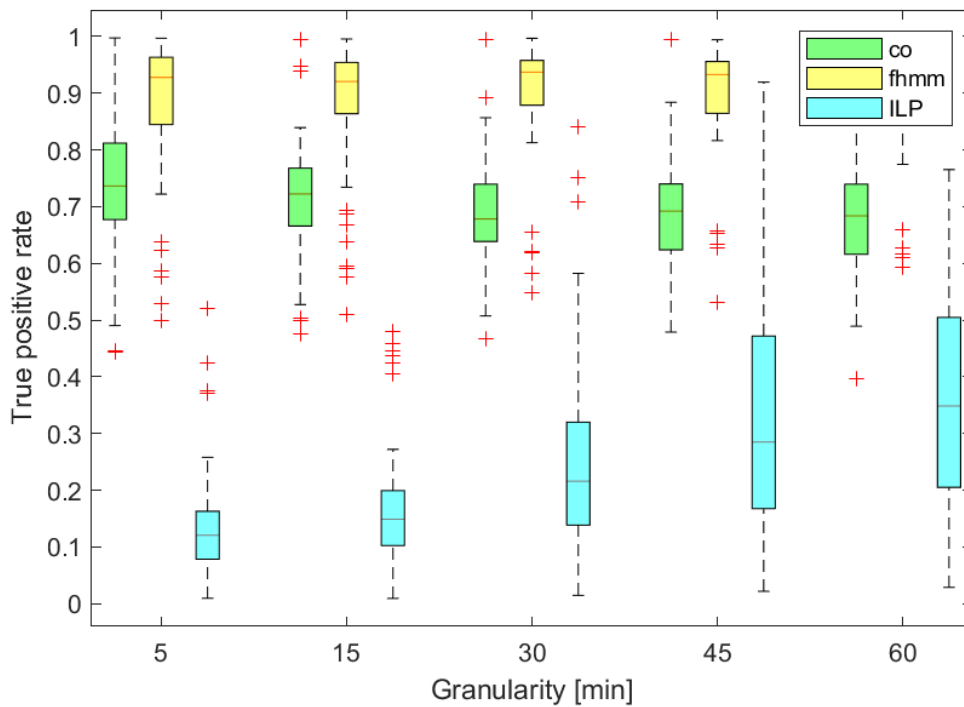


Figure 6: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

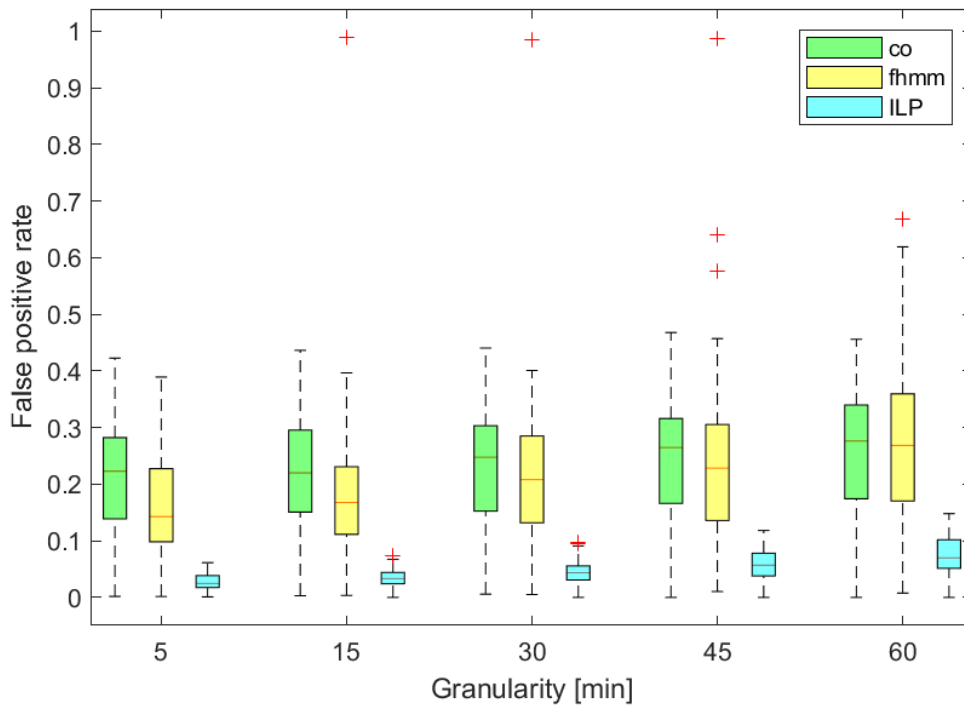


Figure 7: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

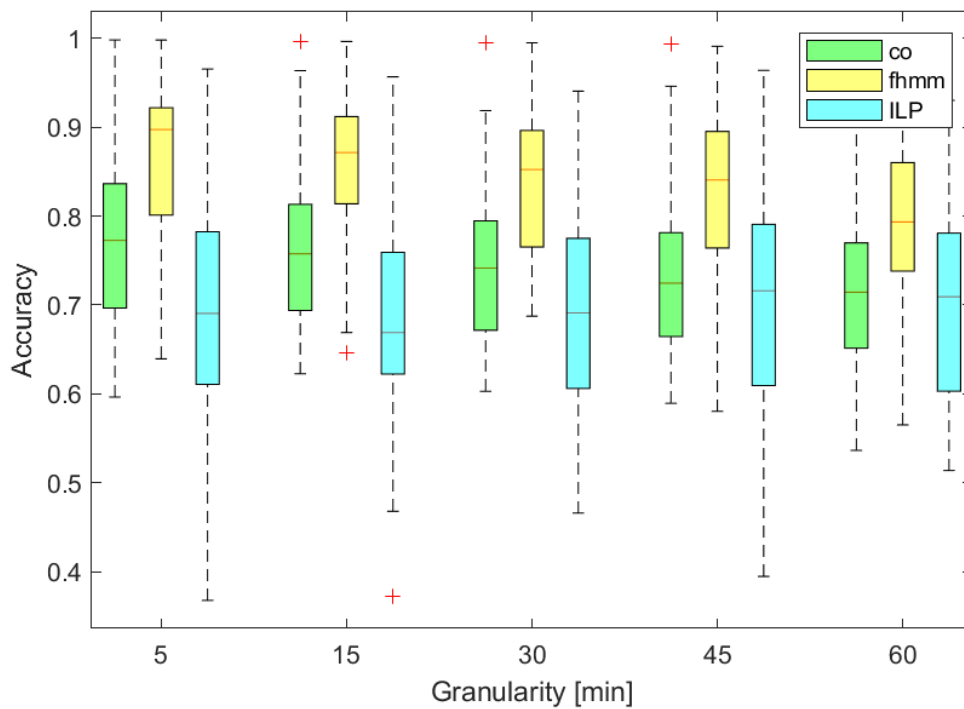


Figure 8: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

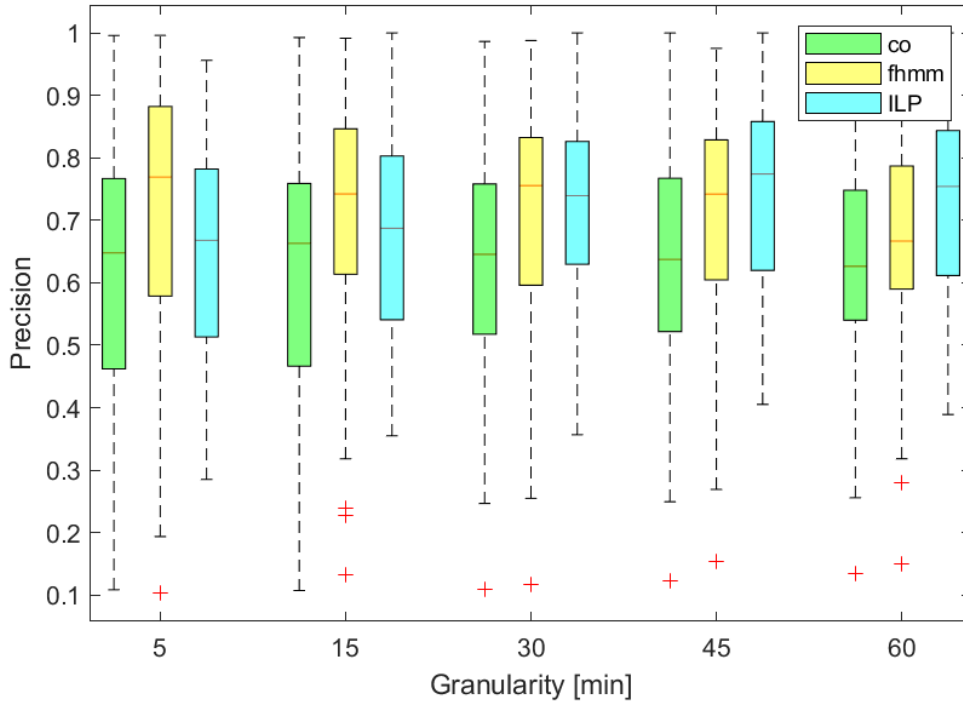


Figure 9: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the synthetic dataset, for different measurement granularities.

Results shows that the fraction of energy consumption correctly disaggregated and the root mean square error achieved by the ILP algorithm are very similar to that achieved by the CO and the FHMM algorithms (see Figure 3 and Figure 5), whereas the ILP algorithm outperforms the CO and FHMM algorithms in terms of normalized error in assigned power (see Figure 4). Note that both the root mean square error and the normalized error tend to decrease when coarsening the measurement granularities. Moreover, though the true positive rate obtained by the ILP algorithm is on average lower than that obtained by the CO and FHMM algorithms, with some improvements at 45 and 60 minutes granularities (see Figure 6), the false positive rate is consistently reduced by the ILP algorithm with respect to the CO and FHMM algorithms (i.e., the FP achieved by the ILP algorithm never exceeds 0.1 at 45 and 60 minutes granularity and remains below 0.05 at 5 and 15 mins granularities, whereas the CO and FHMM algorithms have FP on average ranging between 0.2 and 0.3, with peaks up to 0.6). This means that, though the ILP algorithm sometimes does not detect some activity periods of the appliances, it almost never fails in detecting off periods, whereas the CO and FHMM algorithms often incorrectly turns on appliances). Overall, the accuracy achieved by the ILP algorithm is comparable to that of the CO and FHMM algorithms (see Figure 8), whereas the precision obtained by the ILP algorithm is higher than that obtained by the CO and FHMM algorithms at coarse granularities (see Figure 9 at 45 and 60 minutes epochs).

We then repeat the analysis considering the predefined set of appliances defined in Section 4.2. In addition to global metric, we also report the metrics obtained for each category of appliance, excluding the air conditioner, which resulted active only in one of the 65 buildings.

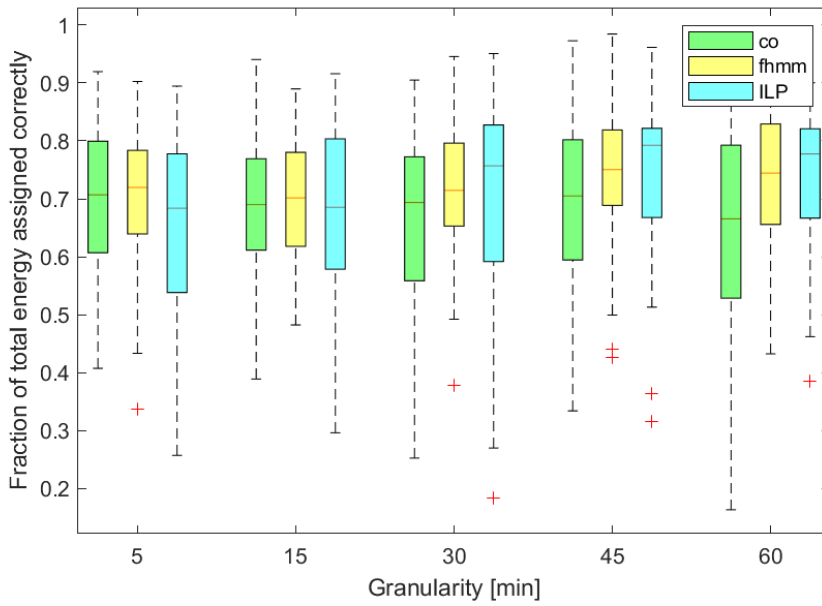


Figure 10: Fraction of Total Energy Assigned Correctly by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

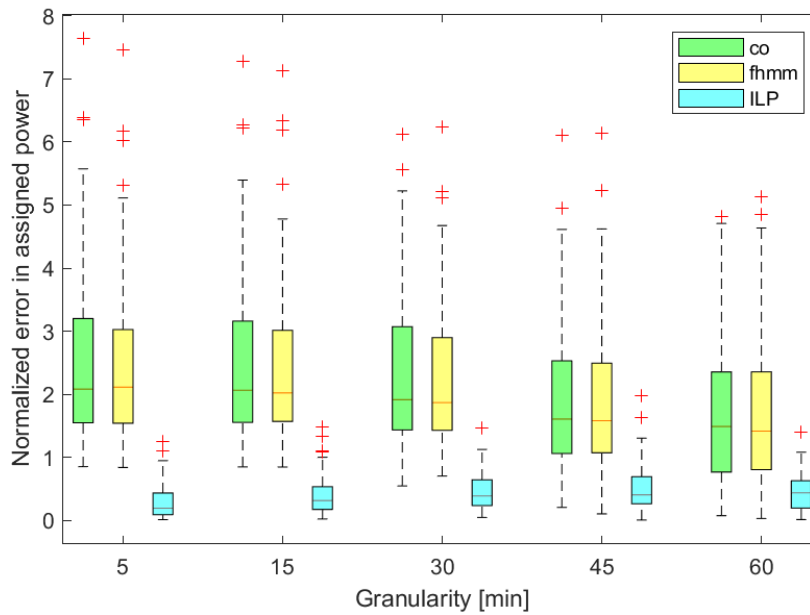


Figure 11: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

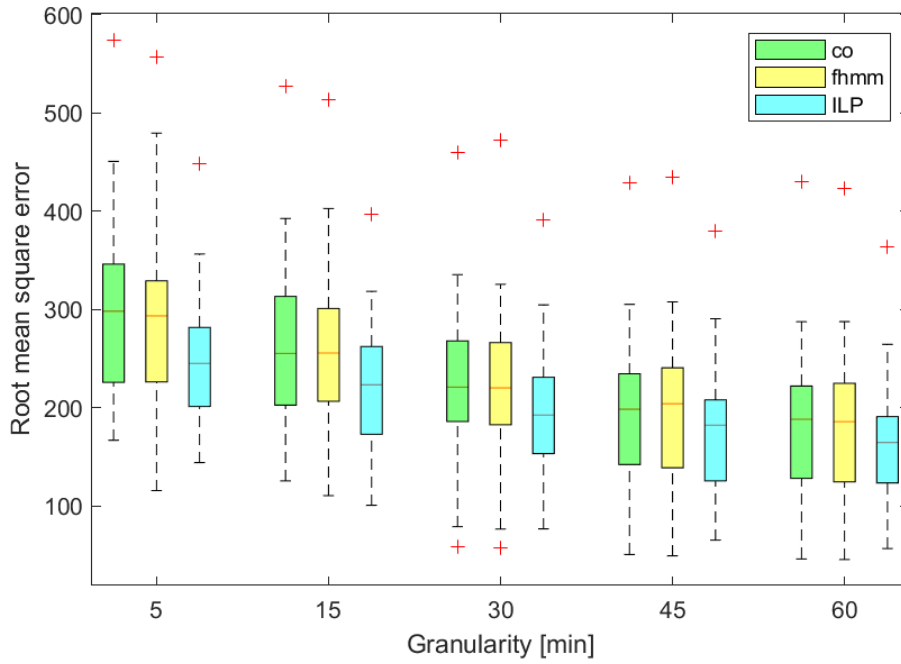


Figure 12: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

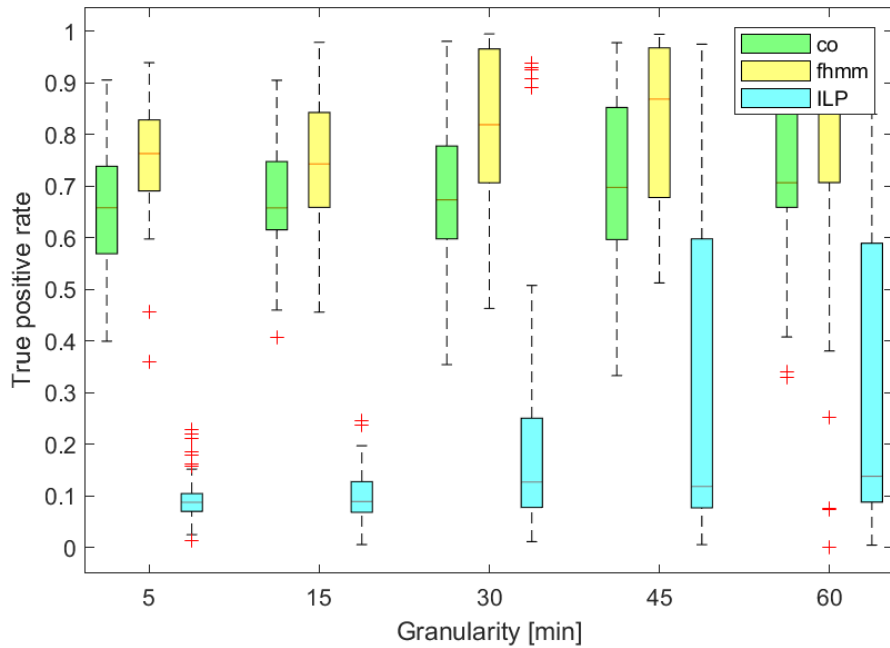


Figure 13: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

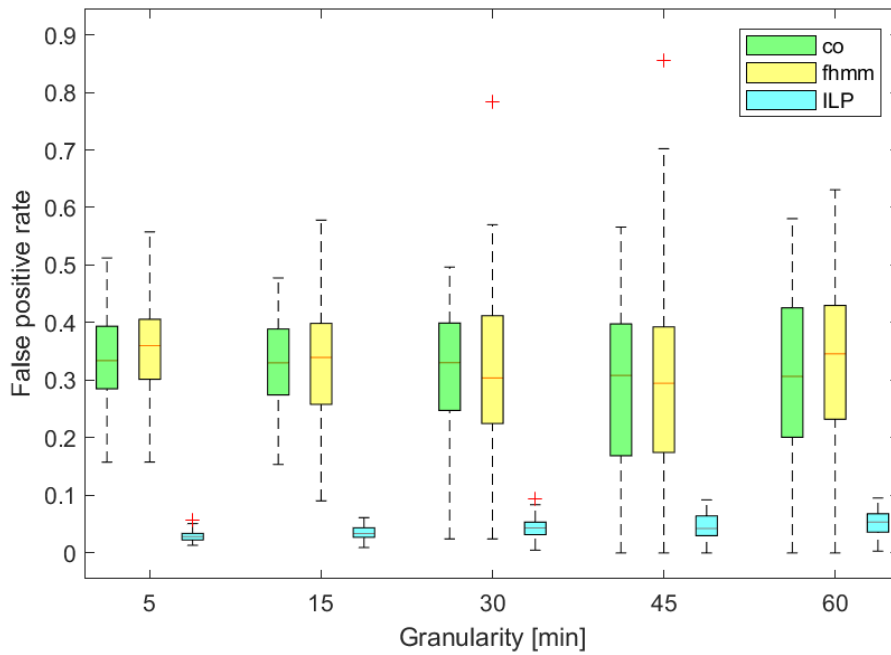


Figure 14: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

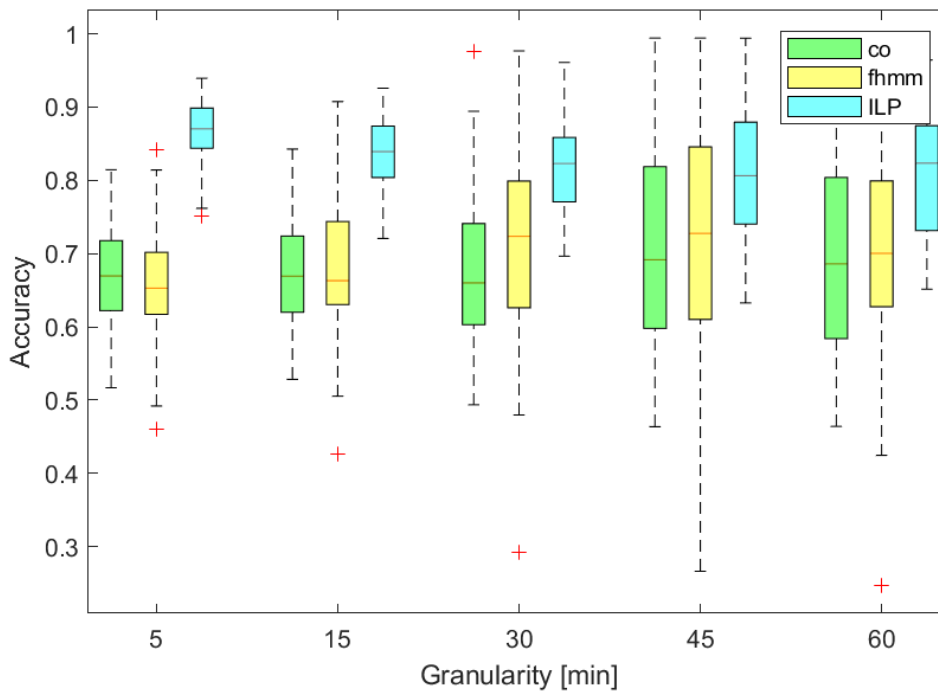


Figure 15: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

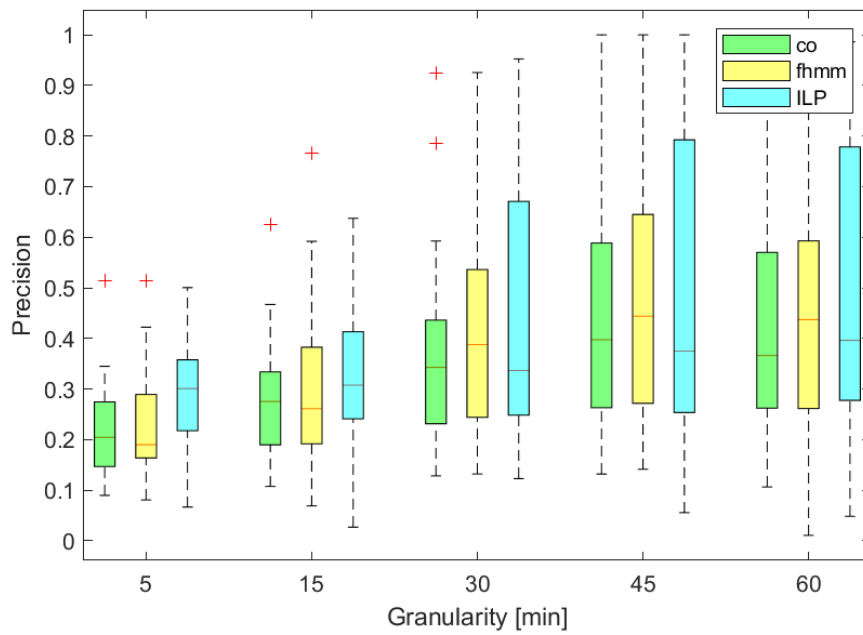


Figure 16: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the synthetic dataset, for different measurement granularities.

The trends of the overall metrics are very similar to those obtained when disaggregating the top five consuming appliances, as reported in Figure 10 to Figure 16, though this time the ILP algorithm outperforms the CO and FHMM algorithms in terms of accuracy at coarse granularities, whereas precision is comparable.

In the following subsections, we comment the values of the performance metrics obtained for single appliance categories.

4.4.1.1 Fridge Disaggregation

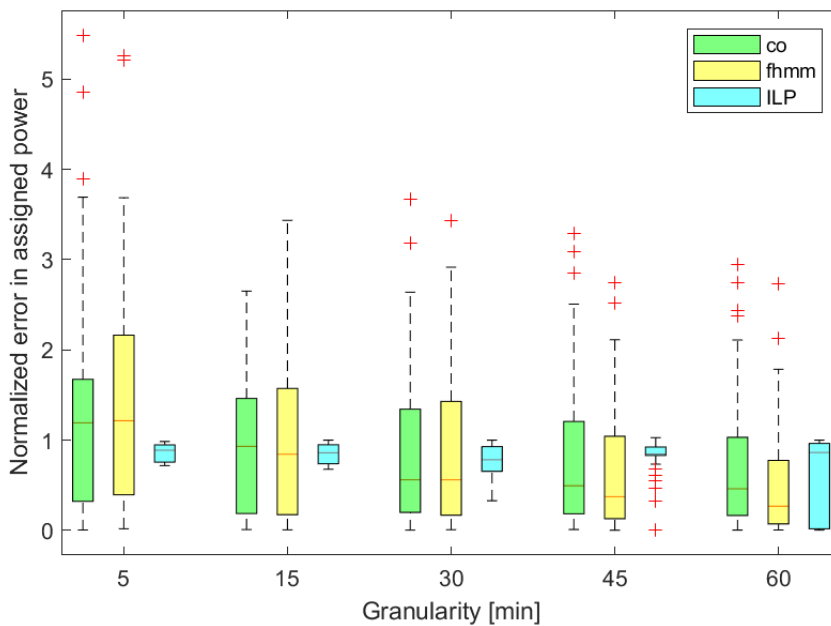


Figure 17: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the fridge in the synthetic dataset, for different measurement granularities.

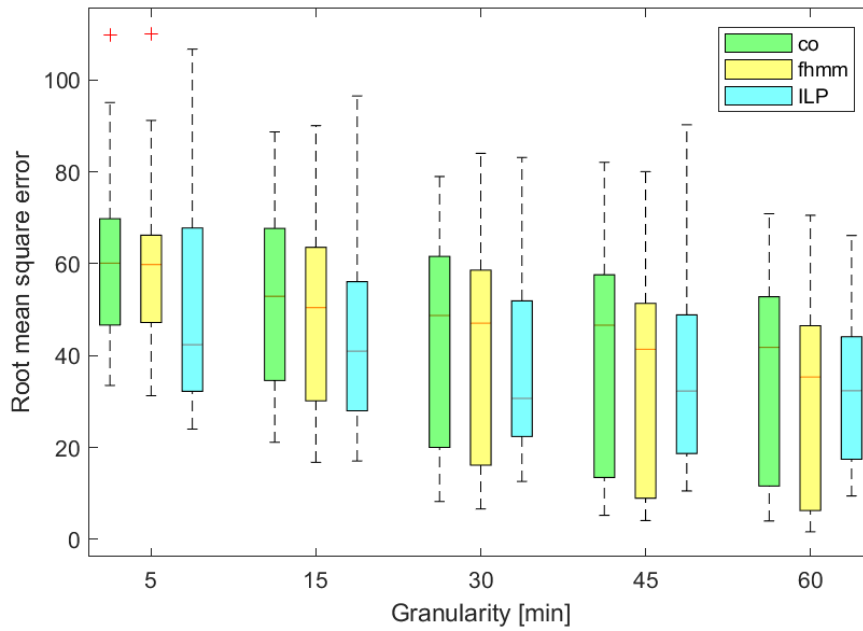


Figure 18: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the fridge in the synthetic dataset, for different measurement granularities.

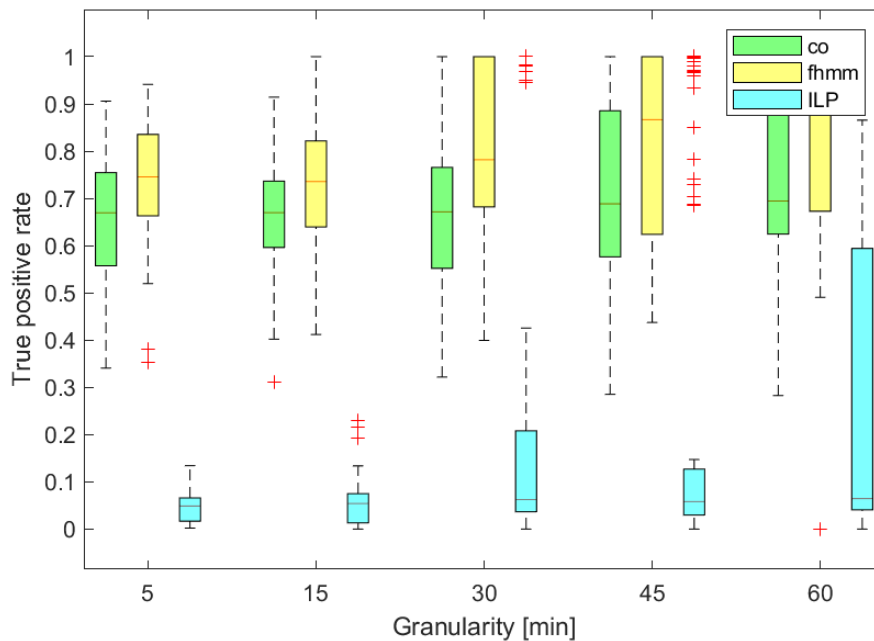


Figure 19: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating fridge in the synthetic dataset, for different measurement granularities.

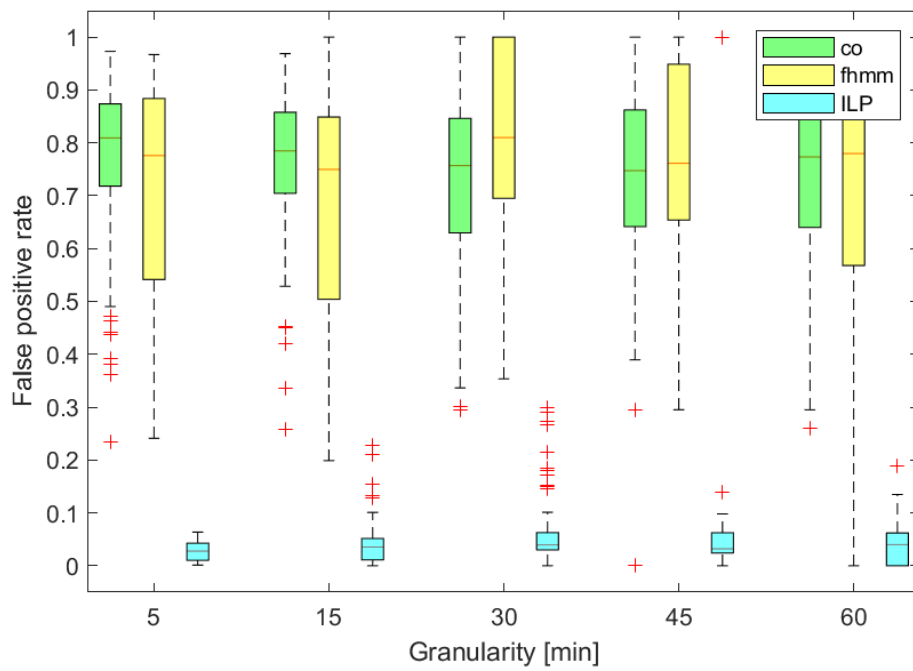


Figure 20: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the fridge in the synthetic dataset, for different measurement granularities.

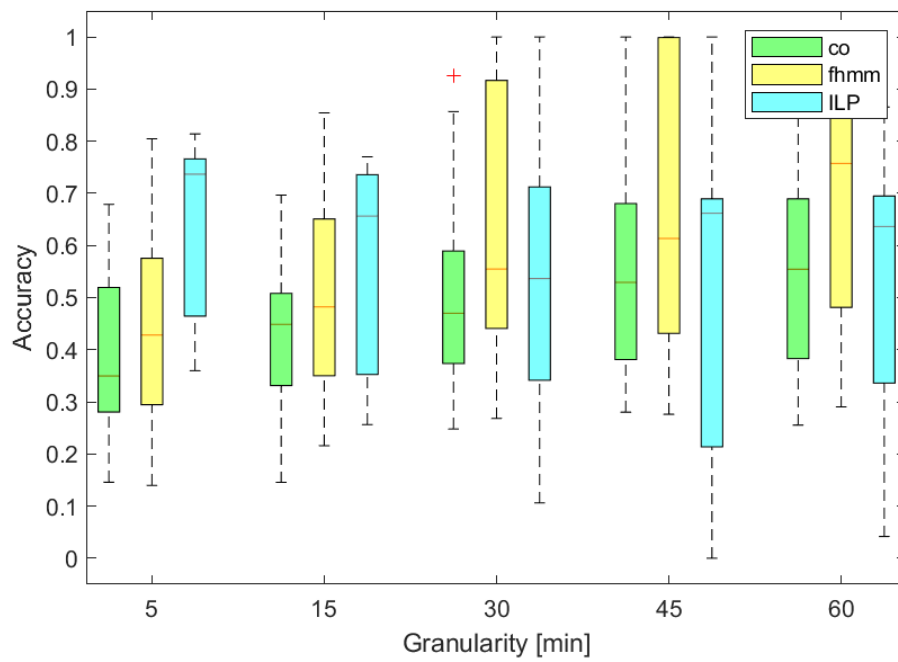


Figure 21: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the fridge in the synthetic dataset, for different measurement granularities.

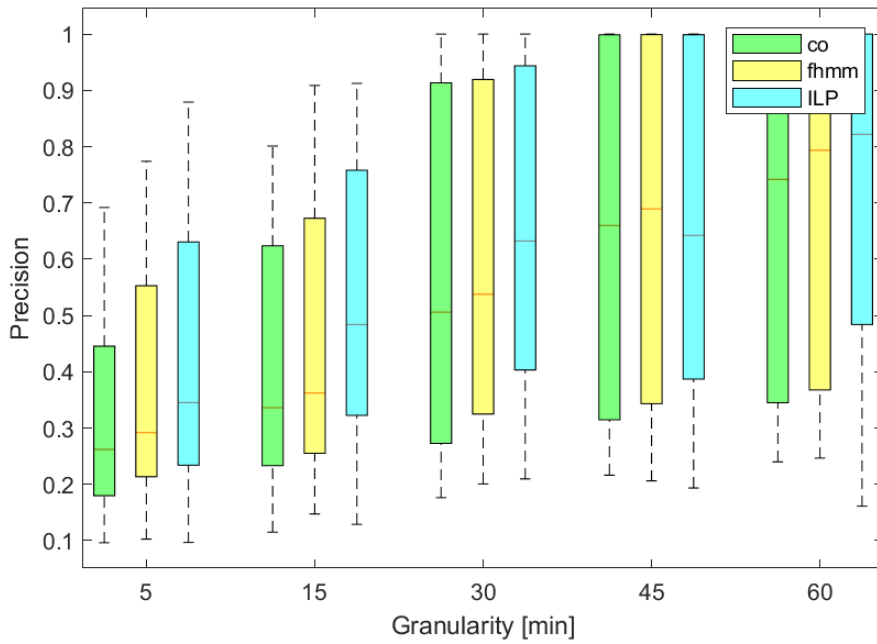


Figure 22: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating the fridge in the synthetic dataset, for different measurement granularities.

Performance figures obtained when disaggregating the fridge consumption patterns are reported from Figure 17 to Figure 22. Root mean square error and normalized error in assigned power obtained with the ILP algorithm are in line with those obtained by the CO and FHMM benchmarks. The same holds for accuracy and precision, though FHMM outperforms CO and ILP in terms of precision at coarse granularities. Though the true positive rate achieved by the ILP algorithm is lower than that of the CO and FHMM algorithms, the CO and FHMM algorithms also show an extremely high false positive rate, whereas that the ILP algorithm closely approaches 0. In other words, the two benchmark algorithms tend to assume that the fridge is always active, thus not recognizing the alternance of activity and inactivity periods that typically characterizes the consumption pattern of such type of appliances. Conversely, the ILP algorithm sometimes fails in identifying activity periods, but almost never fails in identifying inactivity periods.

4.4.1.2 Dishwasher Disaggregation

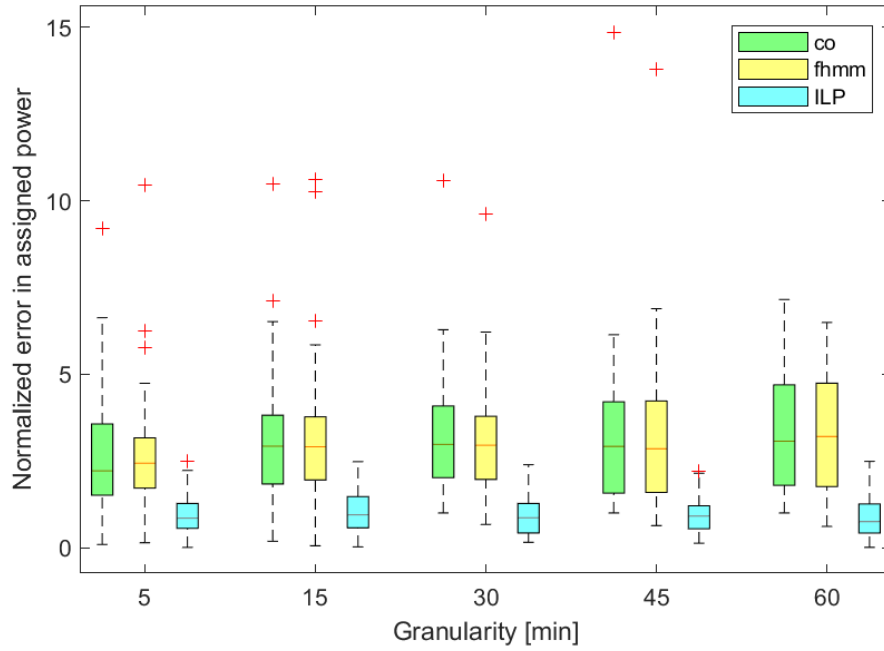


Figure 23: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the dishwasher in the synthetic dataset, for different measurement granularities.

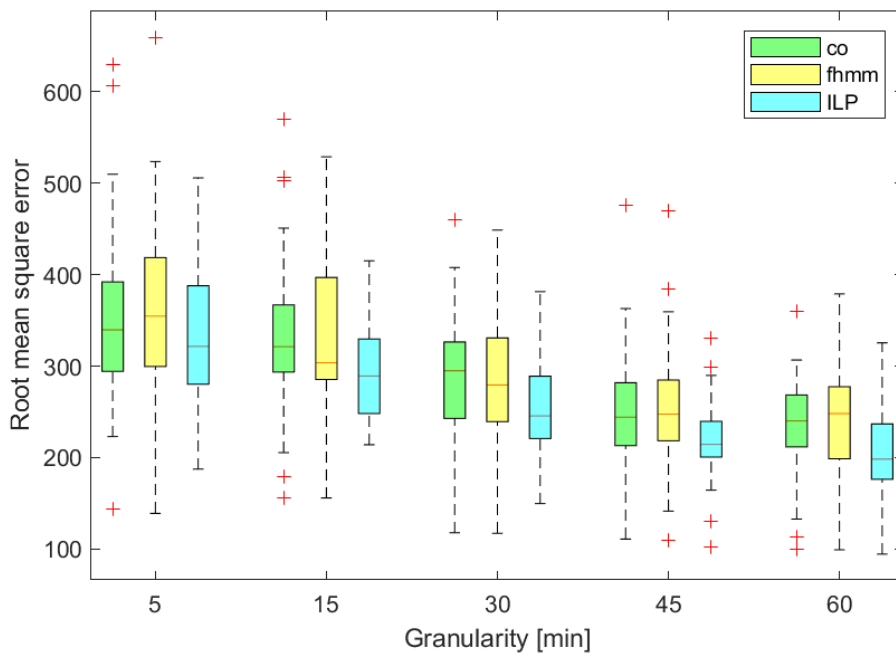


Figure 24: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the dishwasher in the synthetic dataset, for different measurement granularities.

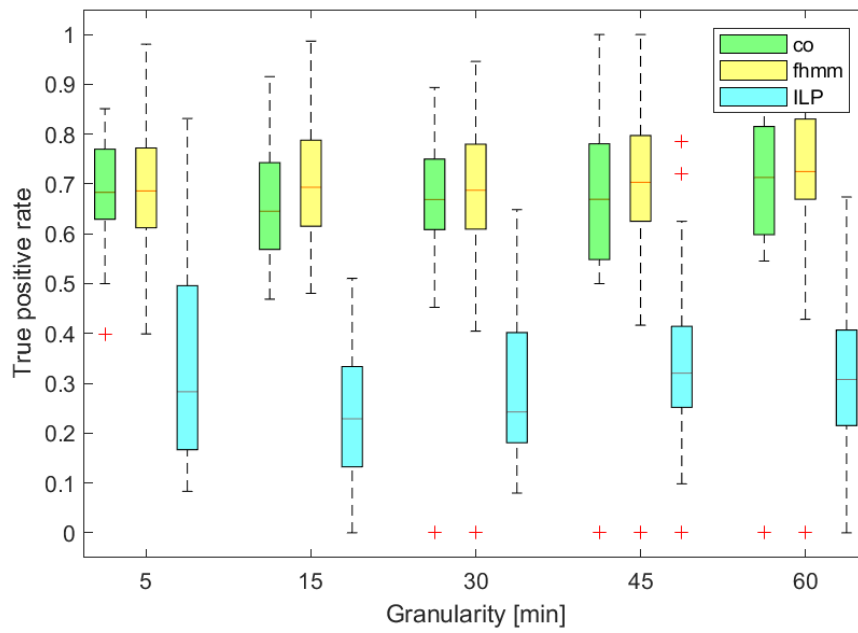


Figure 25: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the dishwasher in the synthetic dataset, for different measurement granularities.

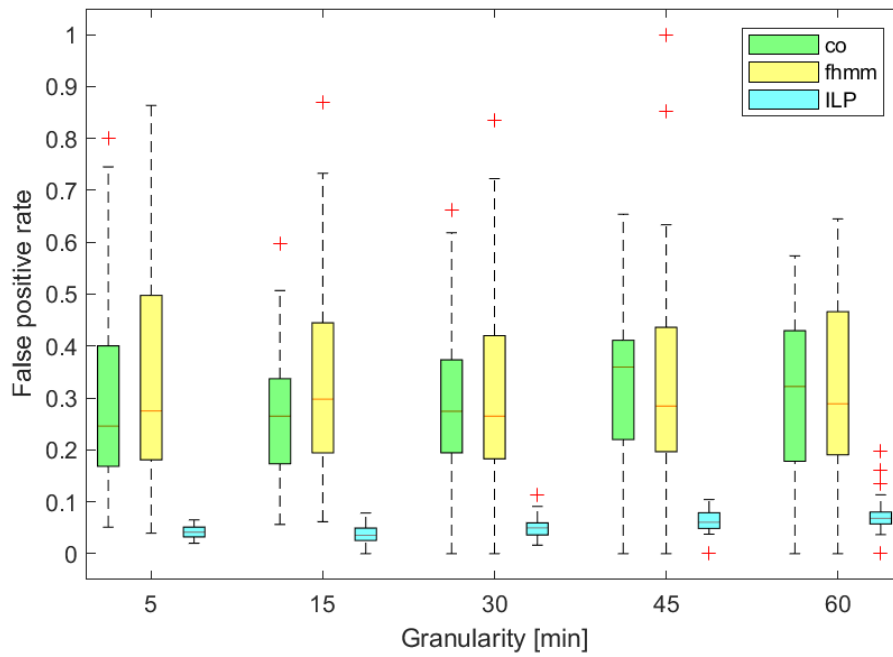


Figure 26: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the dishwasher in the synthetic dataset, for different measurement granularities.

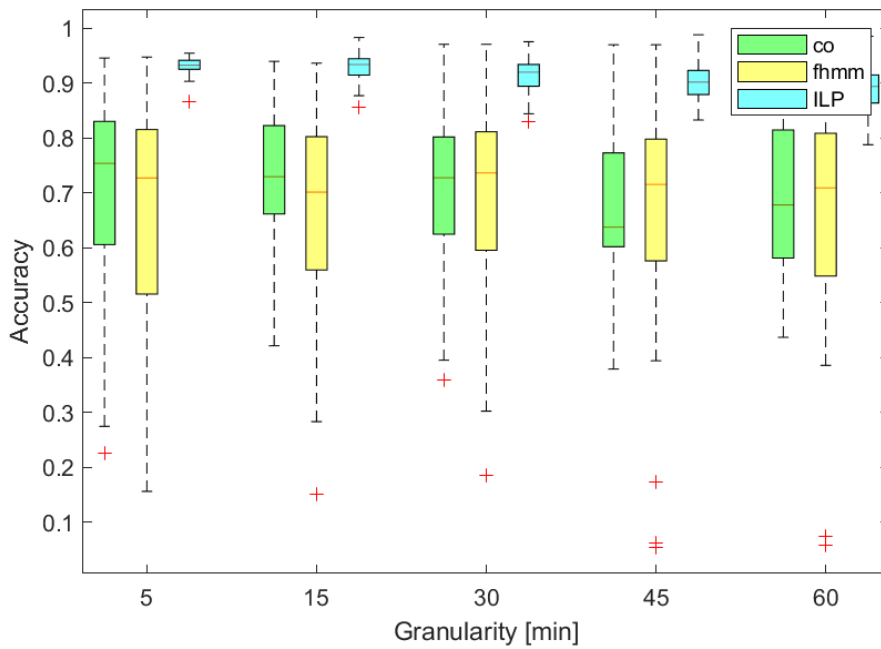


Figure 27: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the dishwasher in the synthetic dataset, for different measurement granularities.

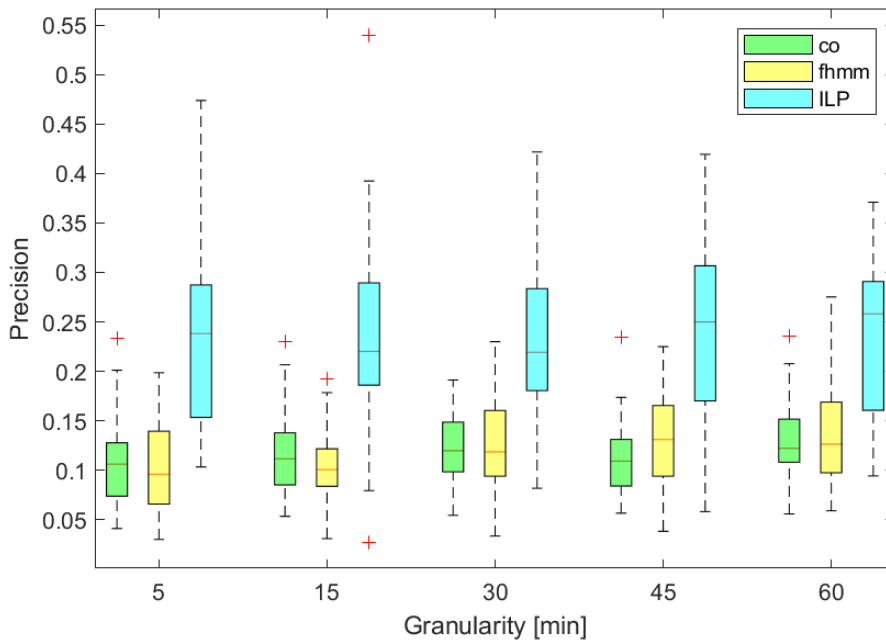


Figure 28: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the synthetic dataset, for different measurement granularities.

Performance figures obtained when disaggregating dishwasher consumption patterns are reported from Figure 23 to Figure 28. The normalized error in assigned power obtained with the ILP algorithm is greatly lower than that obtained by the CO and FHMM benchmarks, whereas the root mean square error of the ILP is on average slightly lower than in the benchmarks. and are in line with those obtained by the CO and FHMM benchmarks. Despite the true positive rate of the ILP approach is lower than that of the benchmarks, the ILP

algorithm outperforms the two benchmarks in terms of accuracy and precision, thanks to its extremely low false positive rate.

4.4.1.3 Electric Oven Disaggregation

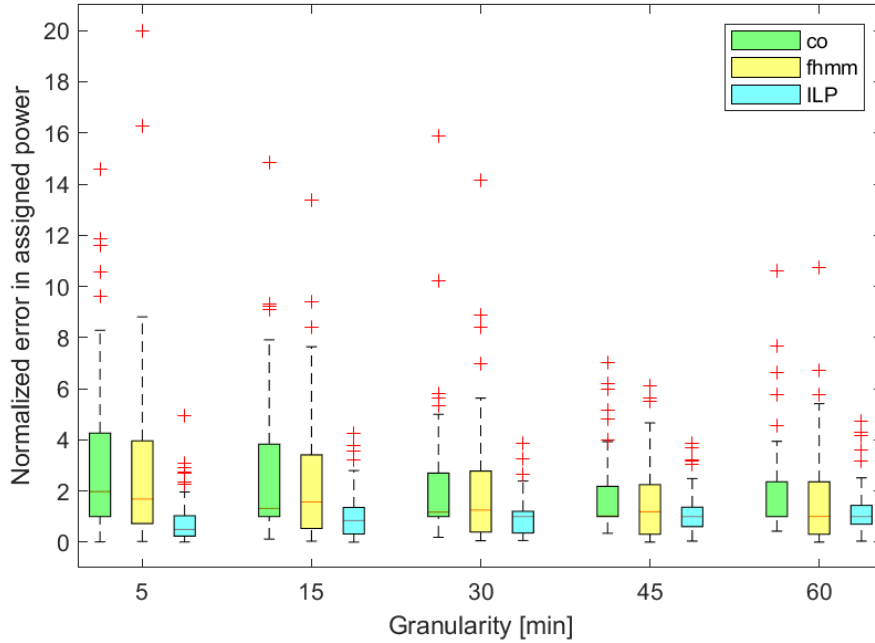


Figure 29: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

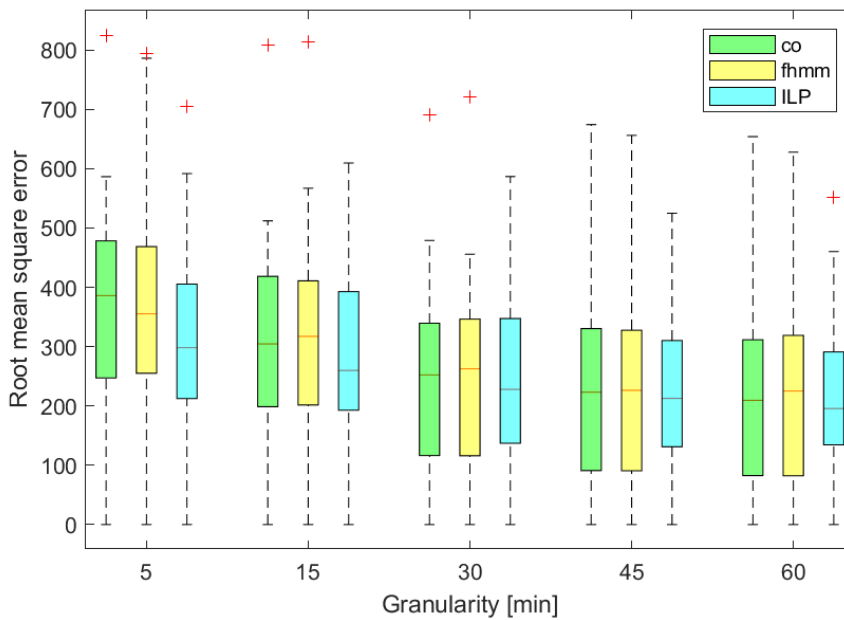


Figure 30: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

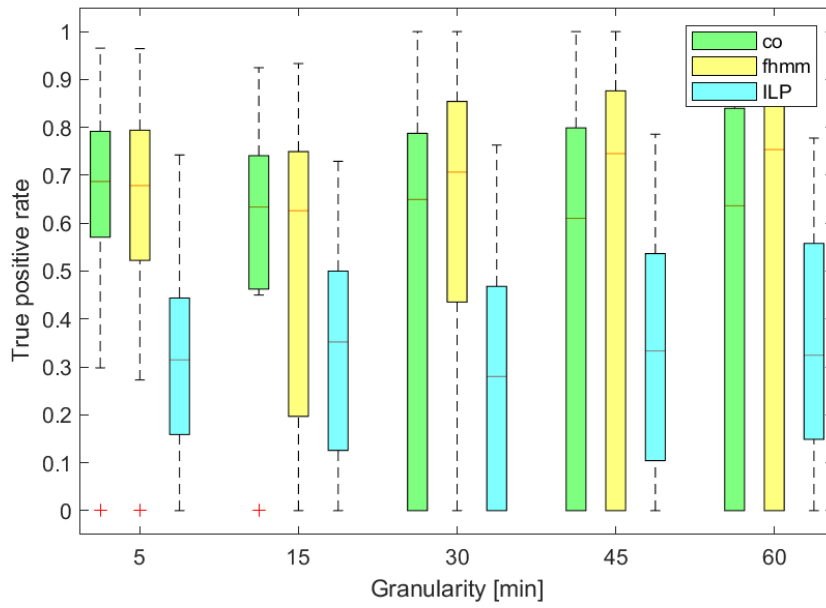


Figure 31: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

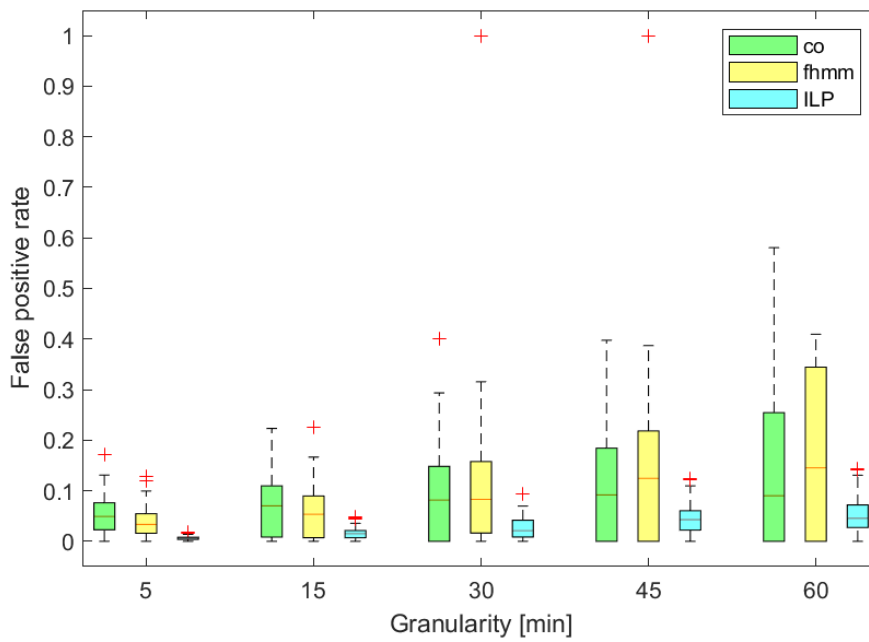


Figure 32: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

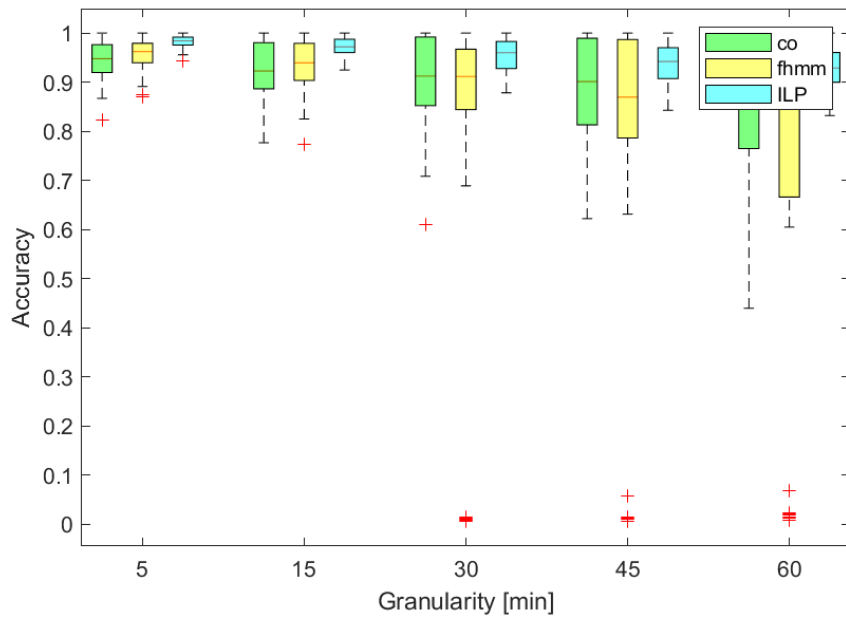


Figure 33: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

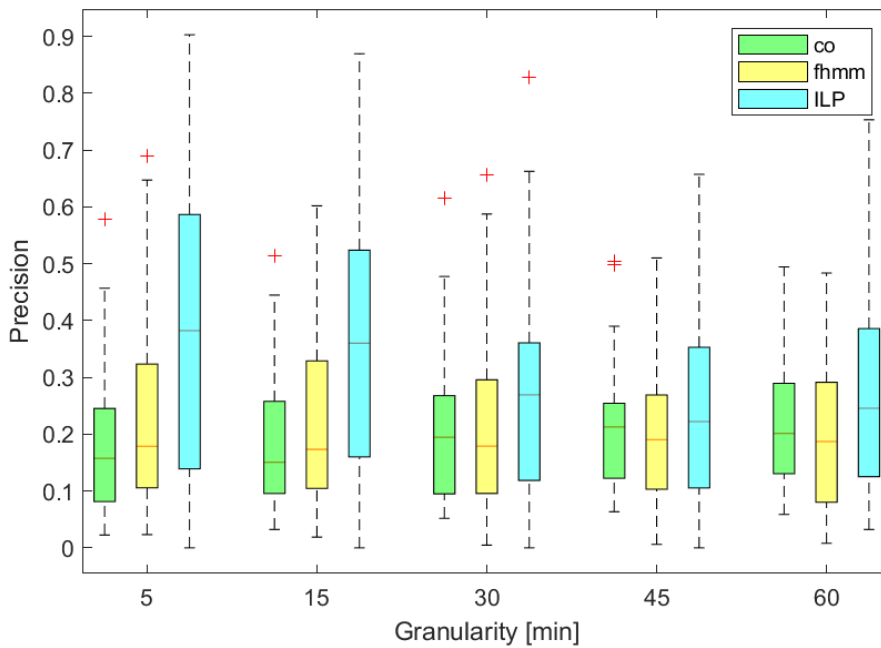


Figure 34: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating the electric oven in the synthetic dataset, for different measurement granularities.

Performance figures obtained when disaggregating electric oven consumption patterns are reported from Figure 29 to Figure 34. In this case, normalized error in assigned power and root mean square error obtained with the ILP algorithm are slightly lower than those obtained by the CO and FHMM benchmarks. The true positive rate of the three algorithms appears to be comparable at coarse granularities, though CO and FHMM exhibit more variability than the ILP algorithm. As usual the ILP algorithm outperforms the two benchmarks

in terms of false positive rate. The accuracy achieved by all algorithms closely approaches 1, whereas the ILP algorithm shows higher precision than CO and FHMM at all time granularities.

4.4.1.4 Washer-Dryer Disaggregation

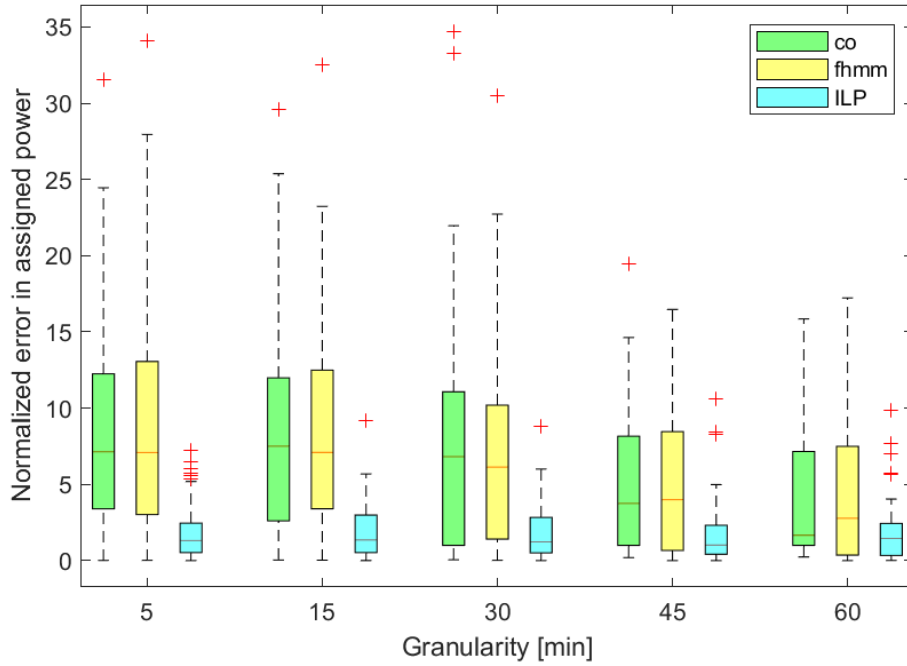


Figure 35: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

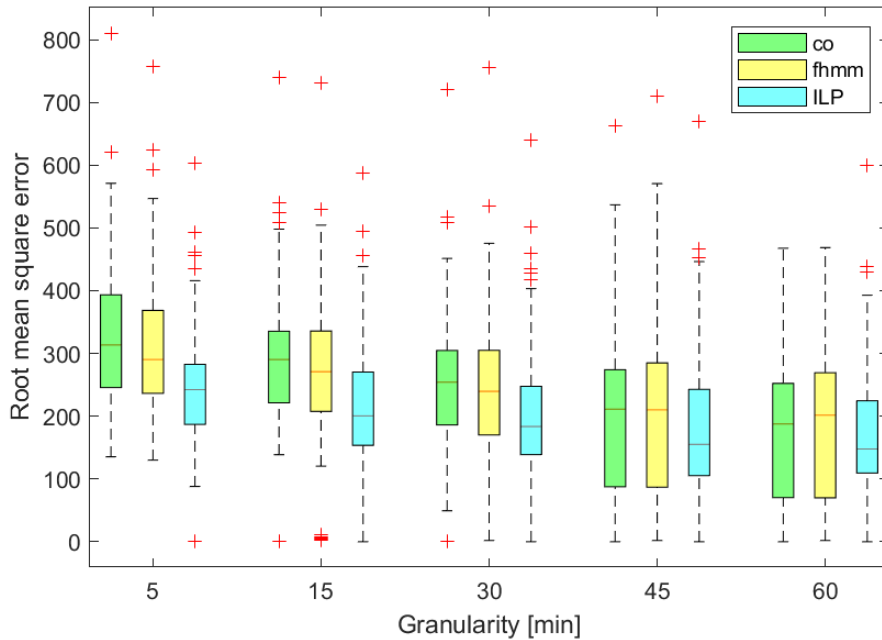


Figure 36: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

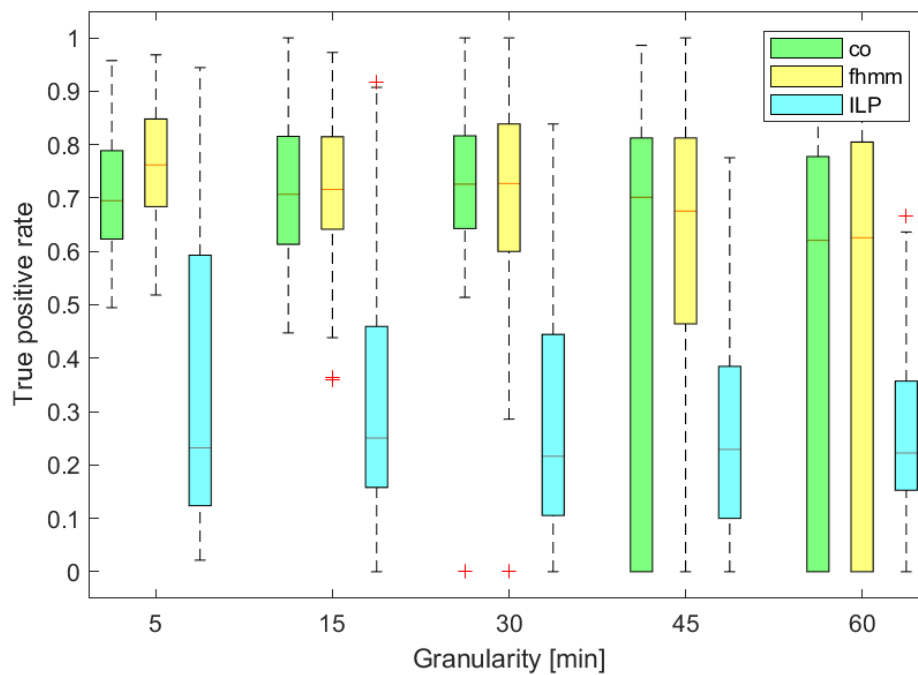


Figure 37: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

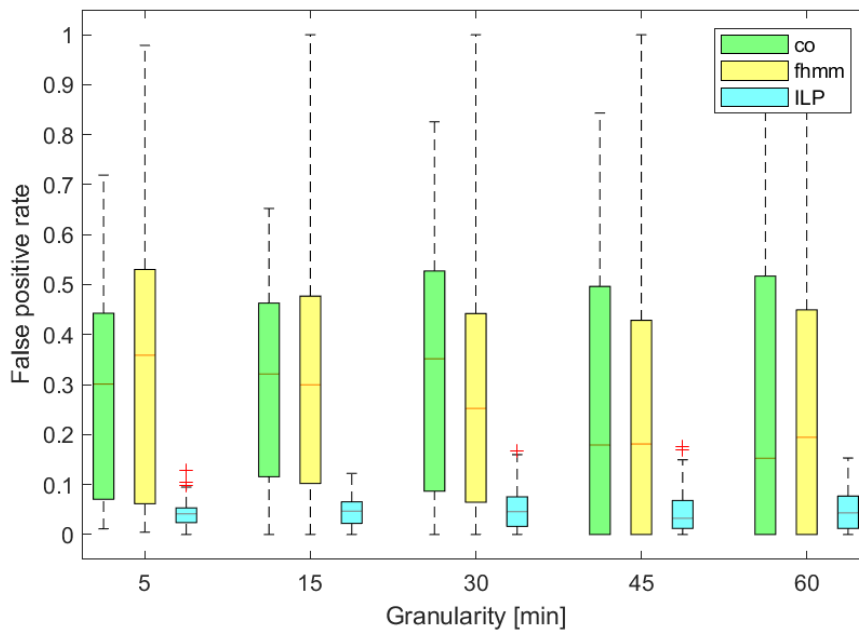


Figure 38: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

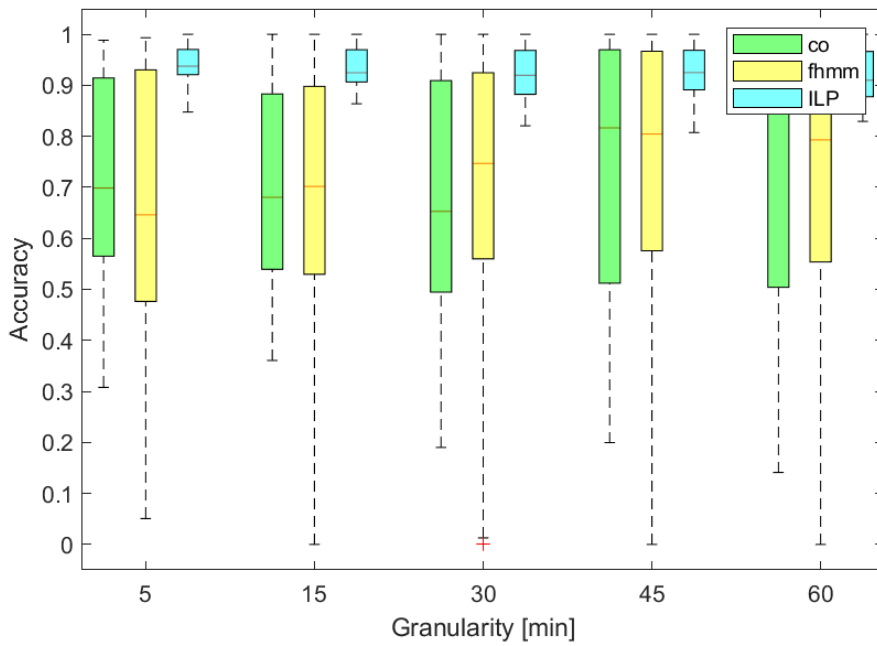


Figure 39: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

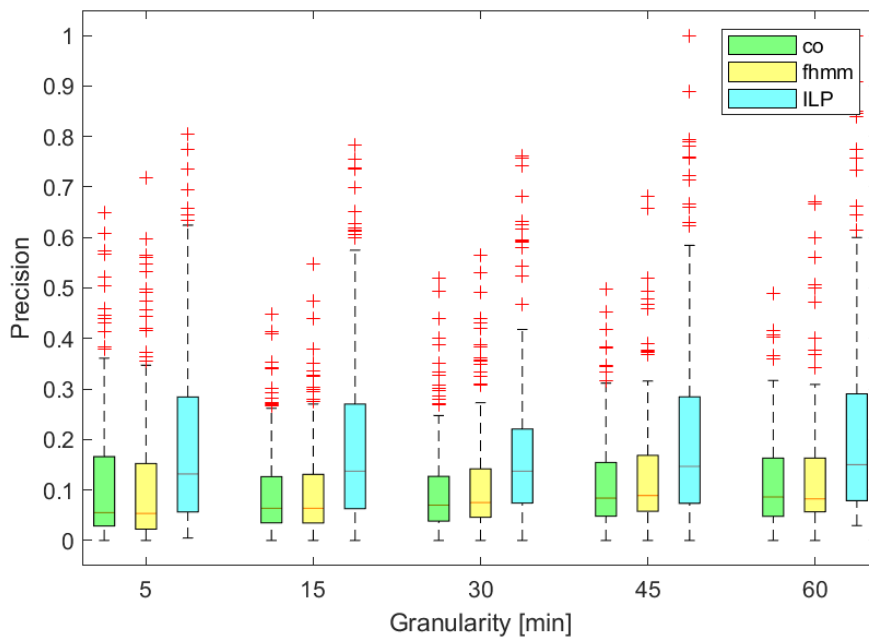


Figure 40: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating the washer-dryer in the synthetic dataset, for different measurement granularities.

Performance figures obtained when disaggregating washer-dryer consumption patterns are reported from Figure 35 to Figure 40. The normalized error in assigned power obtained by the ILP approach is consistently lower than that achieved by CO and FHMM, whereas the root mean square error obtained with the ILP algorithm is slightly lower than those obtained by the CO and FHMM benchmarks. The true positive rate of the ILP algorithm is lower than that of the two benchmarks, but the false positive rate is also consistently lower than in the CO and FHMM (which reach values close to 1, whereas the false positive rate obtained by the ILP

never exceeds 0.2). The accuracy achieved by the ILP algorithm closely approaches 1, thus outperforming the two benchmarks. Though the values of achieved precision are in general quite low, the ILP algorithm shows higher precision than CO and FHMM at all time granularities.

4.4.2 Assessment with UKDale dataset

We now extend our numerical assessment by replicating the analysis of the previous subsection using the UKDale dataset. Again, we report global metrics for the scenario with the top 5 consuming appliances from Figure 41 to Figure 47 and both global and individual metrics for the scenario with a predefined appliance set from Figure 48 to Figure 66.

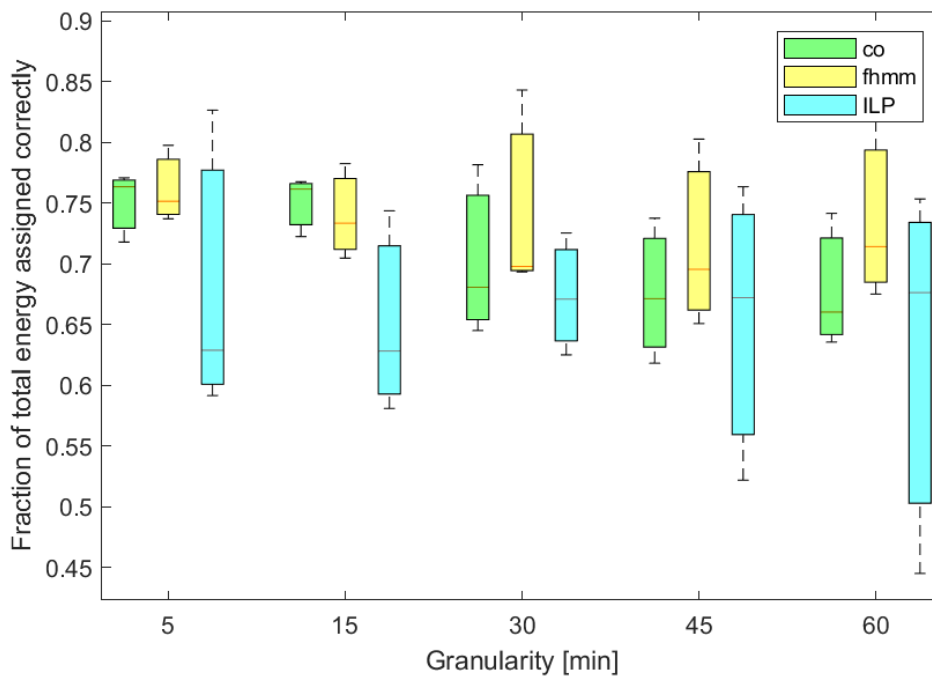


Figure 41: Fraction of Total Energy Assigned Correctly by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

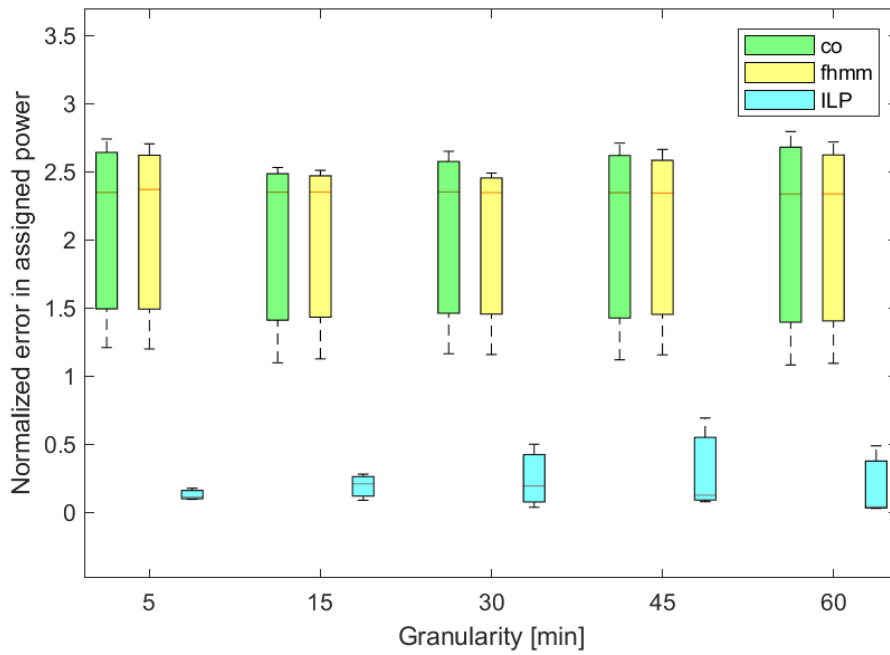


Figure 42: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

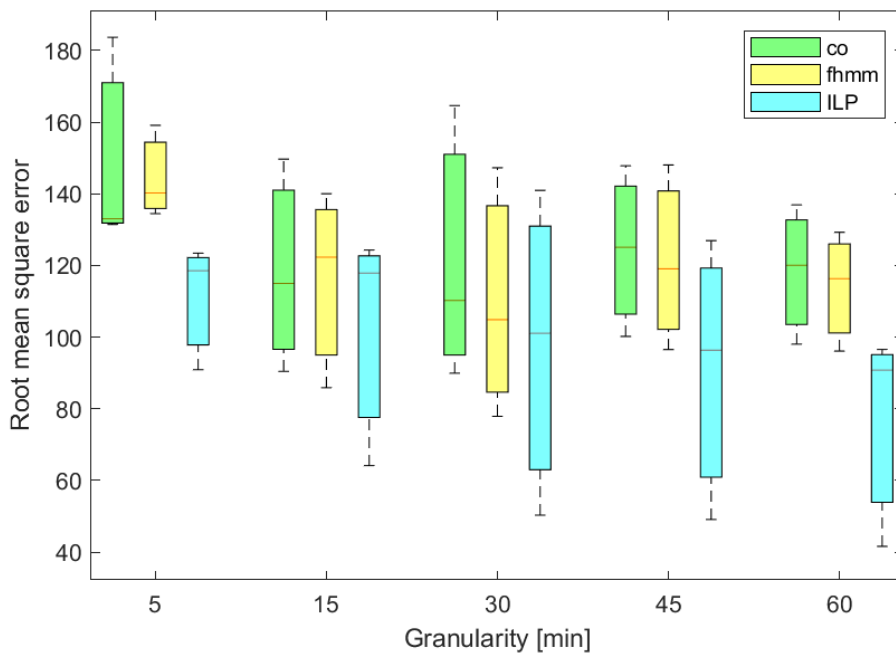


Figure 43: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

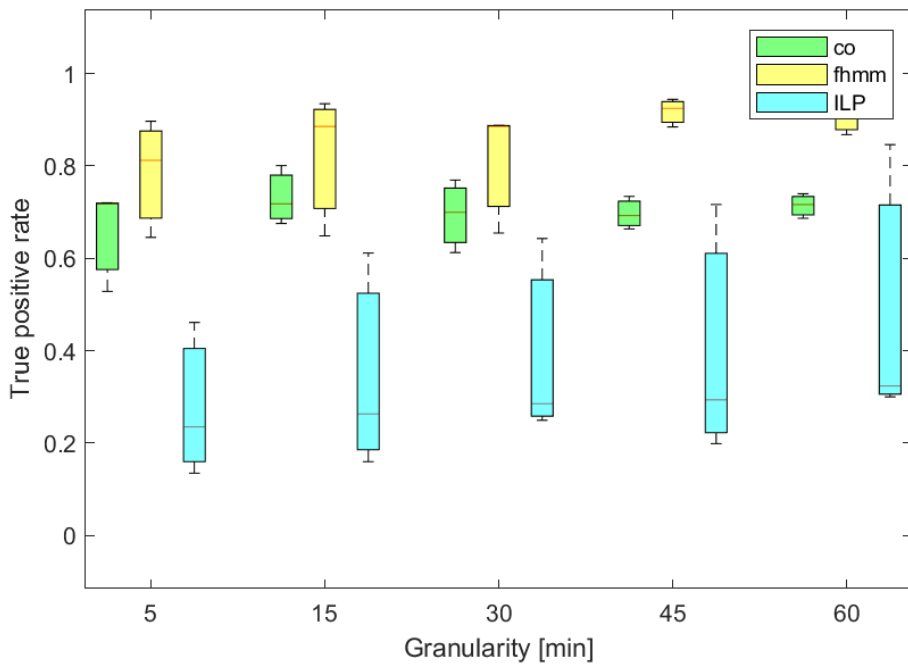


Figure 44: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

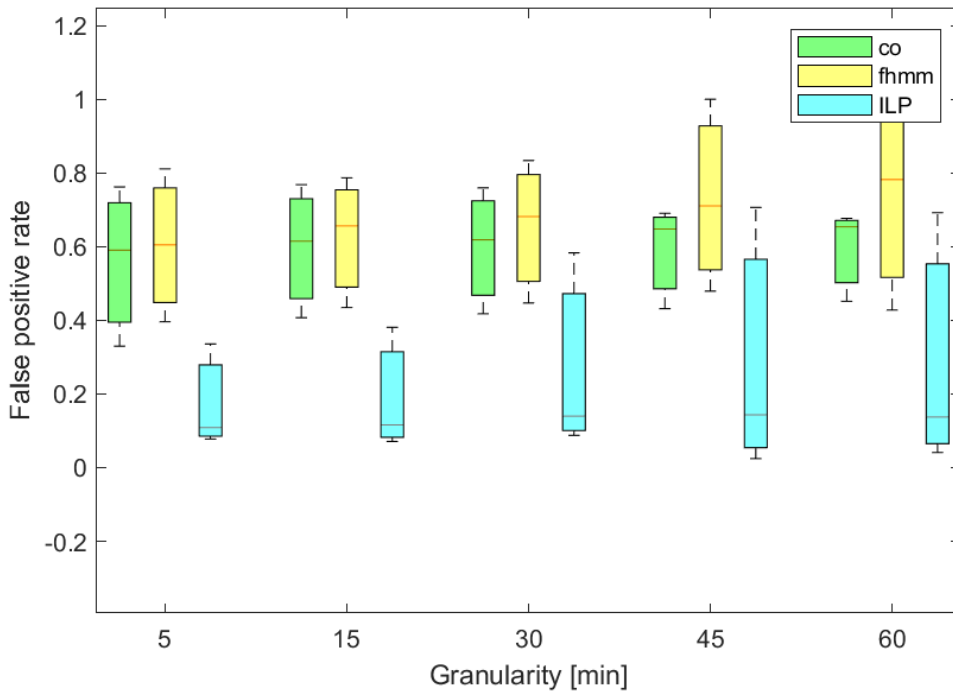


Figure 45: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

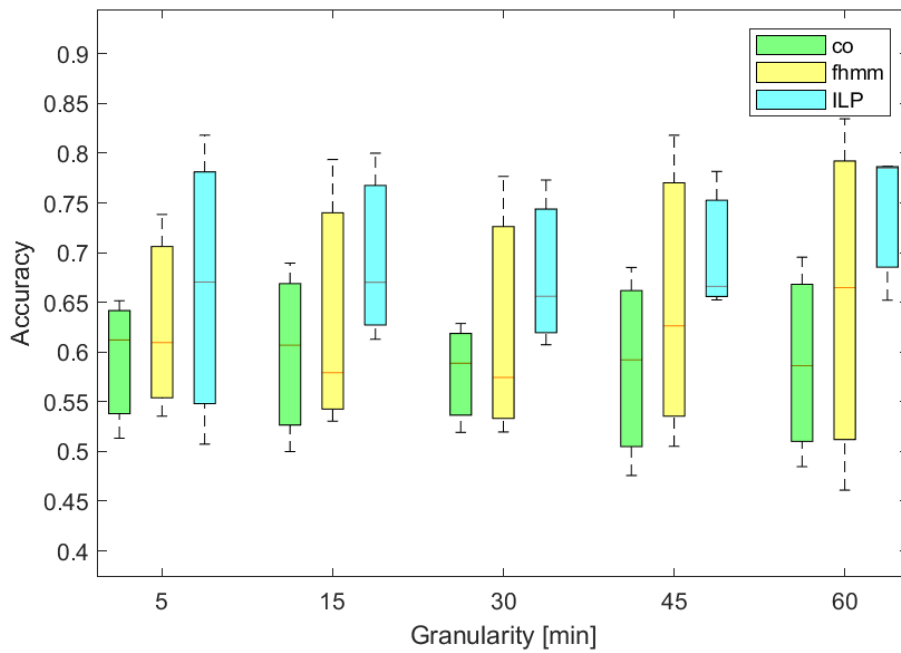


Figure 46: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

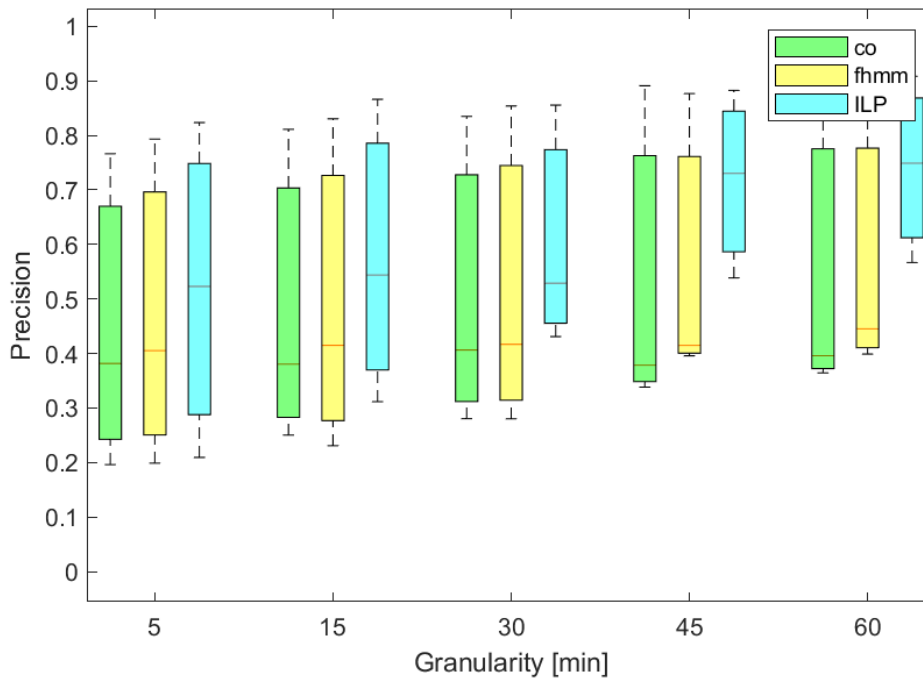


Figure 47: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating the 5 top consuming appliances in the UKDale dataset, for different measurement granularities.

With this dataset, the fraction of energy consumption correctly assigned by the ILP algorithm to the top 5 consuming appliances is slightly lower than that assigned by the CO and FHMM algorithms. However, the normalized error achieved by the ILP algorithm is always greatly smaller than that obtained by the two benchmarks, while the root mean square error achieved by the ILP algorithm is slightly lower than that obtained by CO and FHMM. Similarly to the synthetic dataset, the true positive rate of the ILP algorithm remains lower

than that of the CO and FHMM algorithms, with FHMM outperforming CO. However, an increase in the true positive rate of the ILP algorithm is observed at coarse granularities (45 and 60 minutes epochs). The relatively poor performance of the ILP in terms of true positive rate is compensated by the very low false positive rate, which is greatly smaller than that achieved by the benchmark algorithms. Overall, the ILP algorithm still provides the highest accuracy (especially at coarse granularities) and achieves precision ranges comparable to those of the benchmarks.

We then repeat the analysis considering the predefined set of appliances defined in Section 4.2. In addition to global metrics, we also report the metrics obtained for two category of appliances, namely dishwasher and washer-dryer. Note that none of the three considered buildings of the UKDale dataset was equipped with an air conditioner, whereas individual consumption measurements of fridge and electric oven were available in only one building.

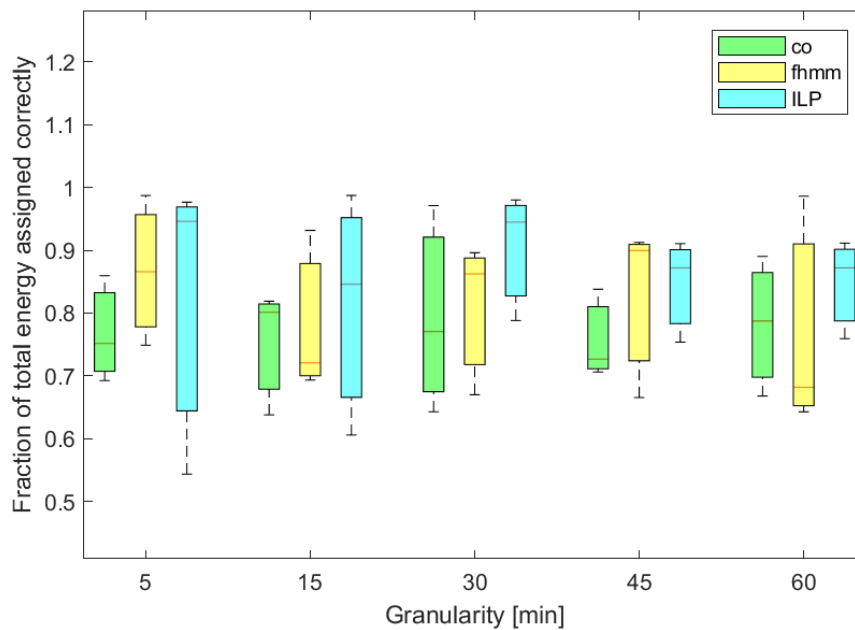


Figure 48: Fraction of Total Energy Assigned Correctly by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

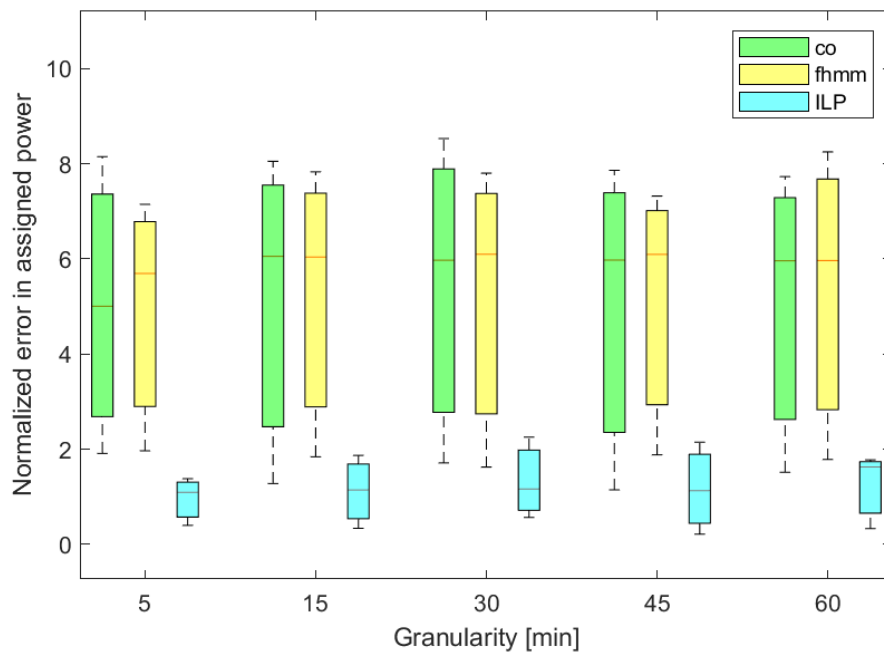


Figure 49: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

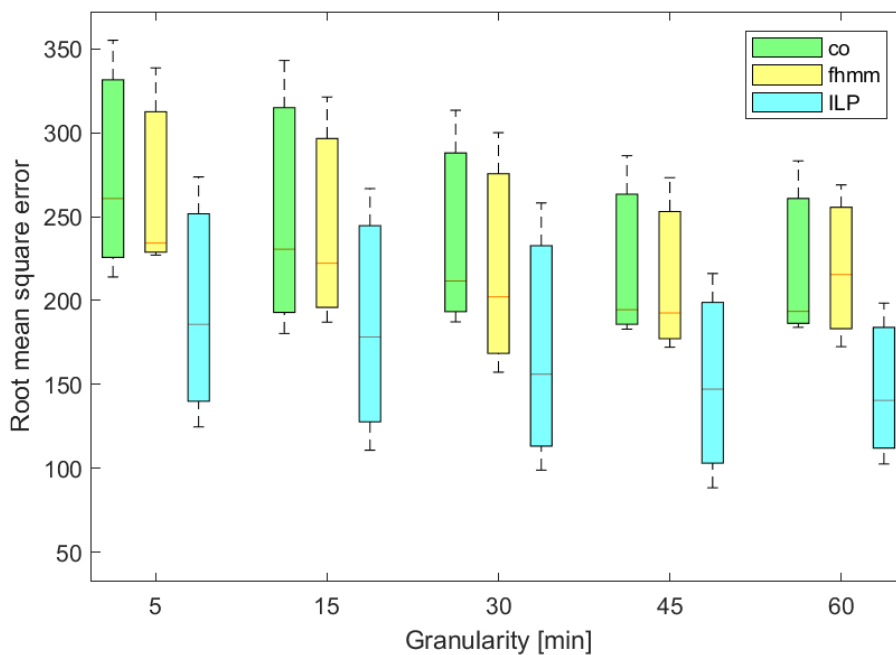


Figure 50: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

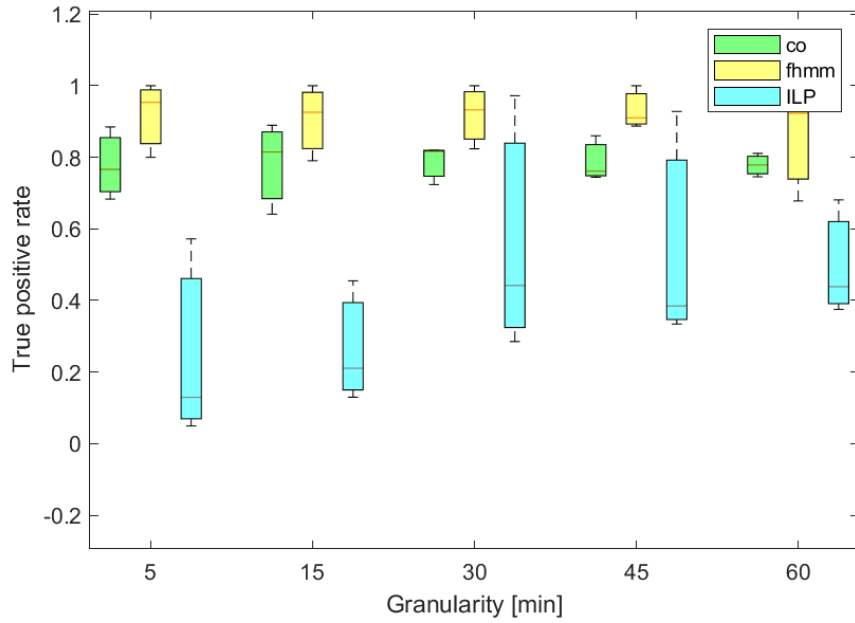


Figure 51: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

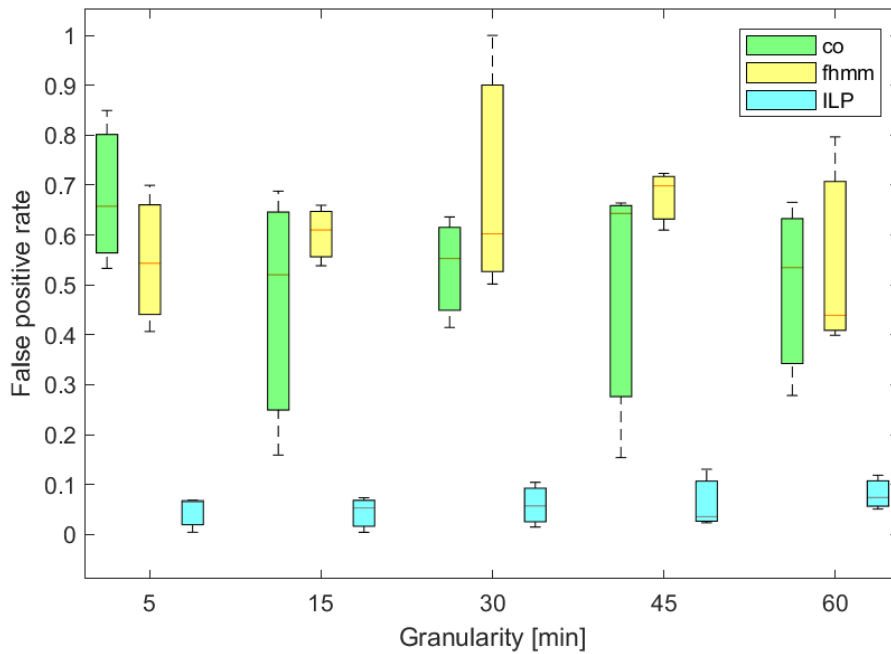


Figure 52: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

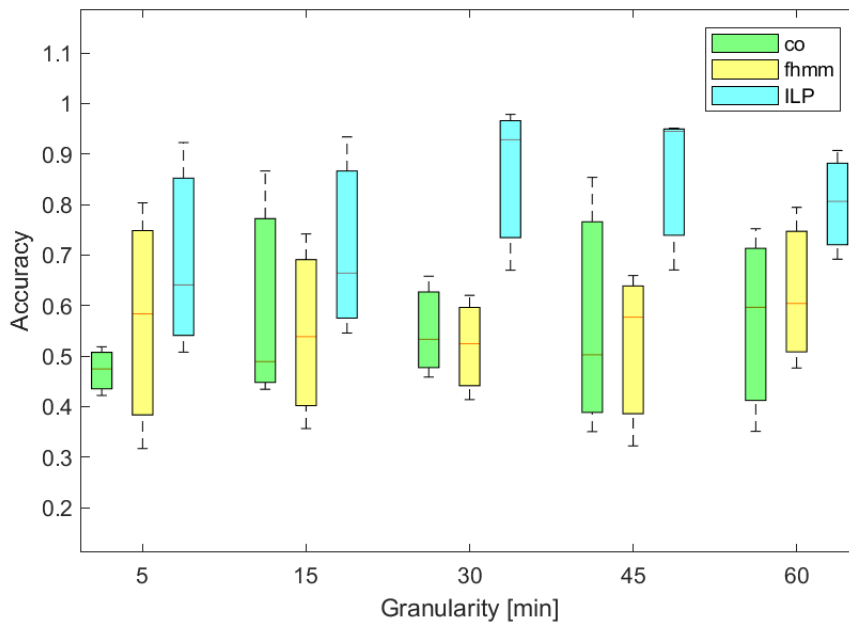


Figure 53: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

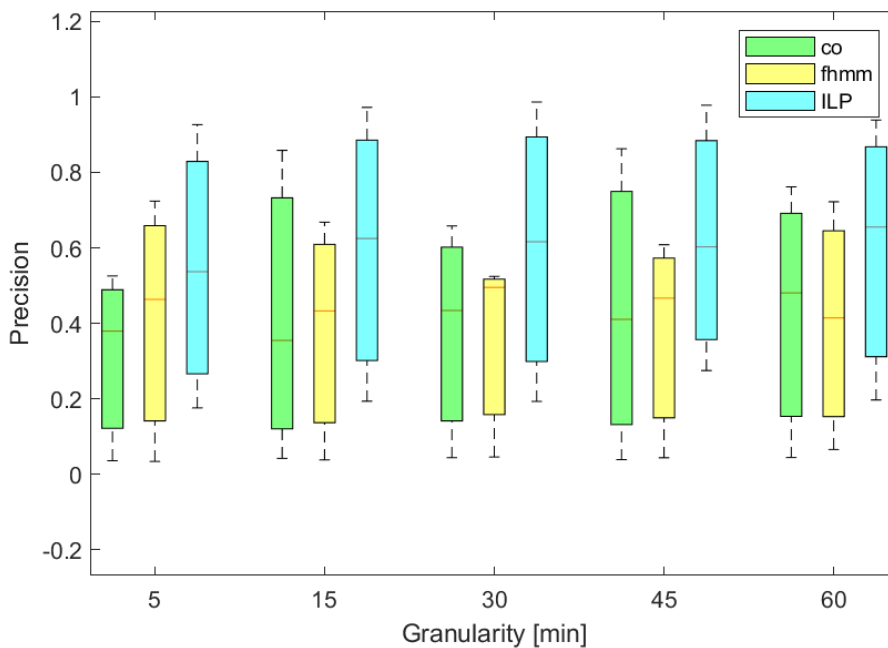


Figure 54: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating disaggregating fridge, dishwasher, electric oven, air conditioner and washer-dryer in the UKDale dataset, for different measurement granularities.

According to the results plotted from Figure 48 to Figure 54 in this setting the fraction of energy consumption correctly assigned by the three algorithms is comparable, whereas ILP consistently outperforms CO and FHMM in terms of normalized and root mean square error. Again, the true positive rate of the ILP algorithm remains lower than that of the CO and FHMM algorithms, with FHMM outperforming CO, though an increase in the true positive rate of the ILP algorithm can be seen at coarse granularities. Also here, the ILP ensures very low false positive rate, outperforming the two benchmark algorithms (with FHMM providing the highest

false positive rate). It follows that the ILP algorithm provides the highest accuracy (especially at coarse granularities) and achieves precision ranges slightly higher than those of the benchmarks.

4.4.2.1 Dishwasher Disaggregation

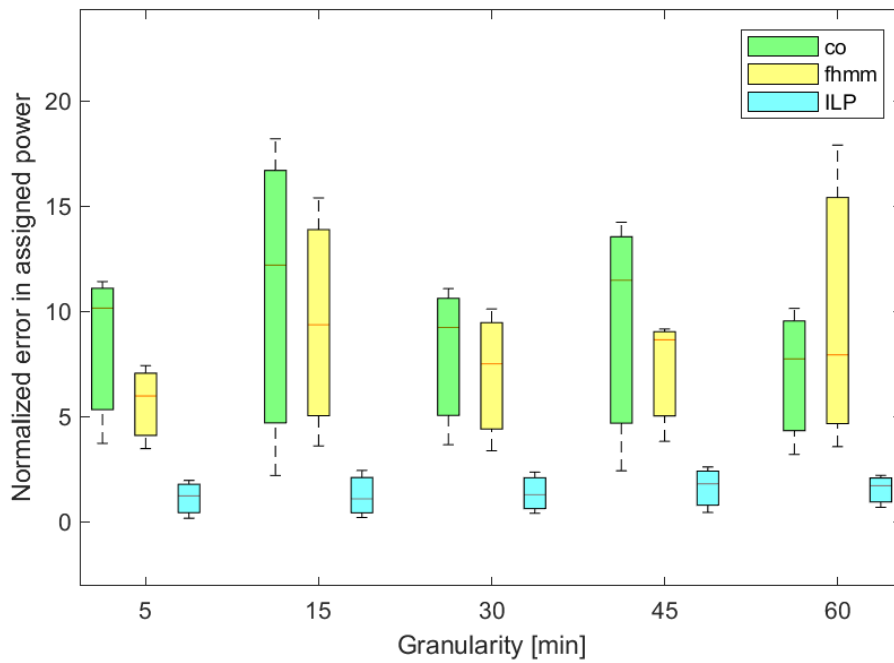


Figure 55: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

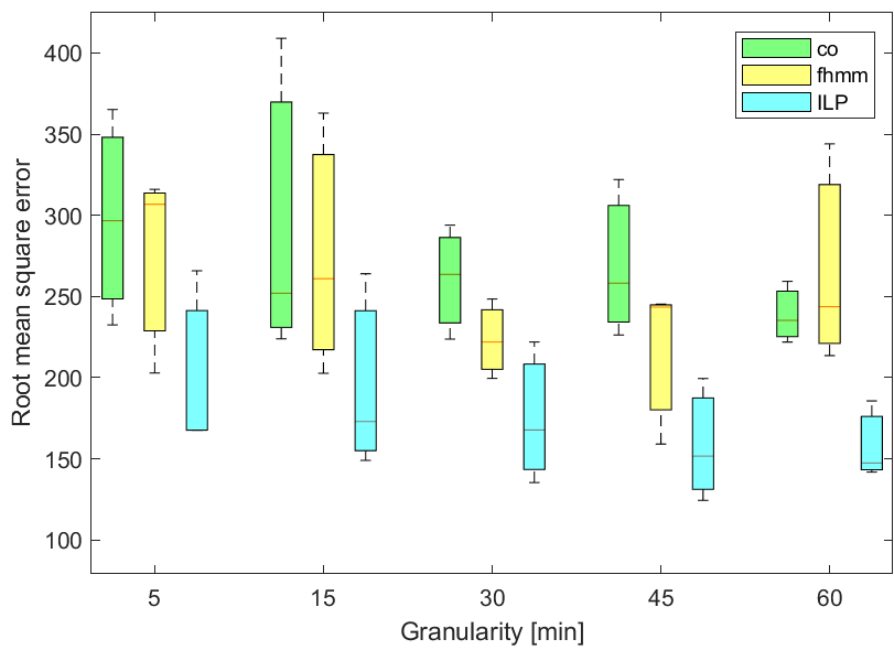


Figure 56: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

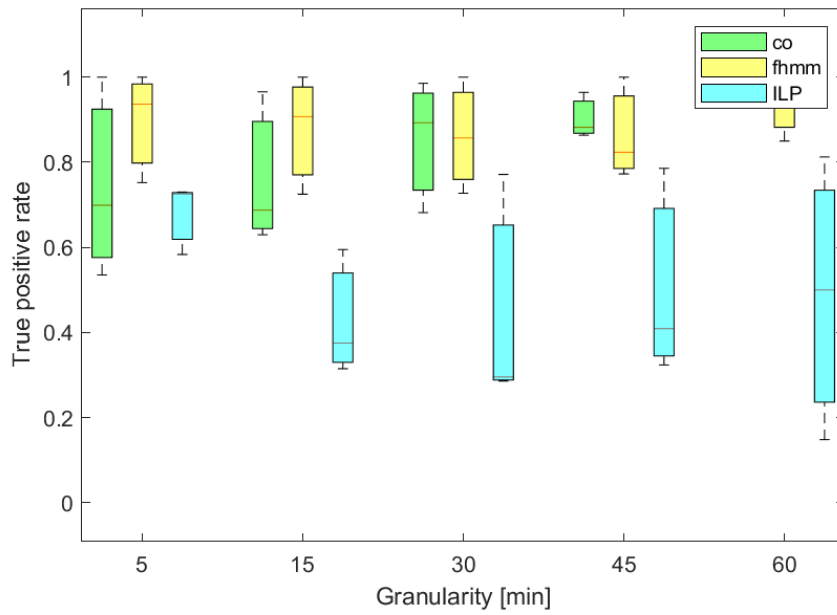


Figure 57: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

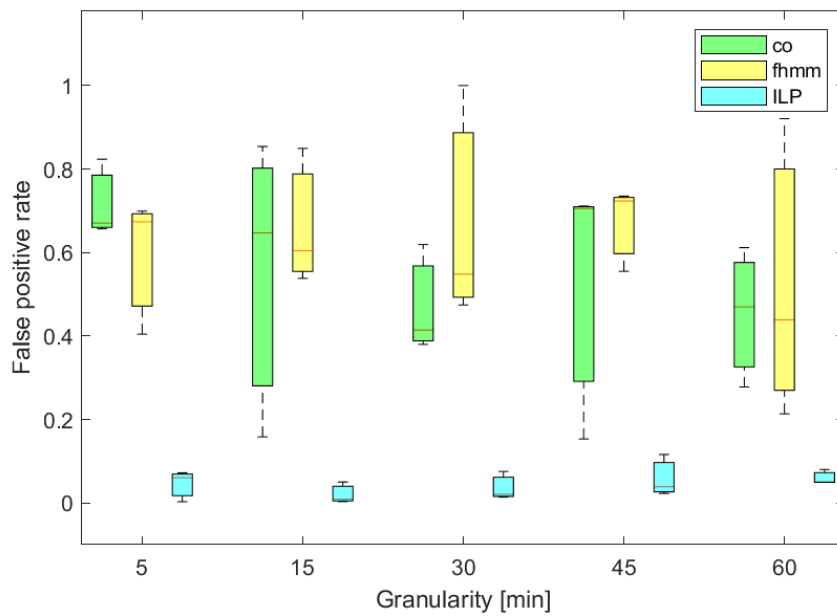


Figure 58: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

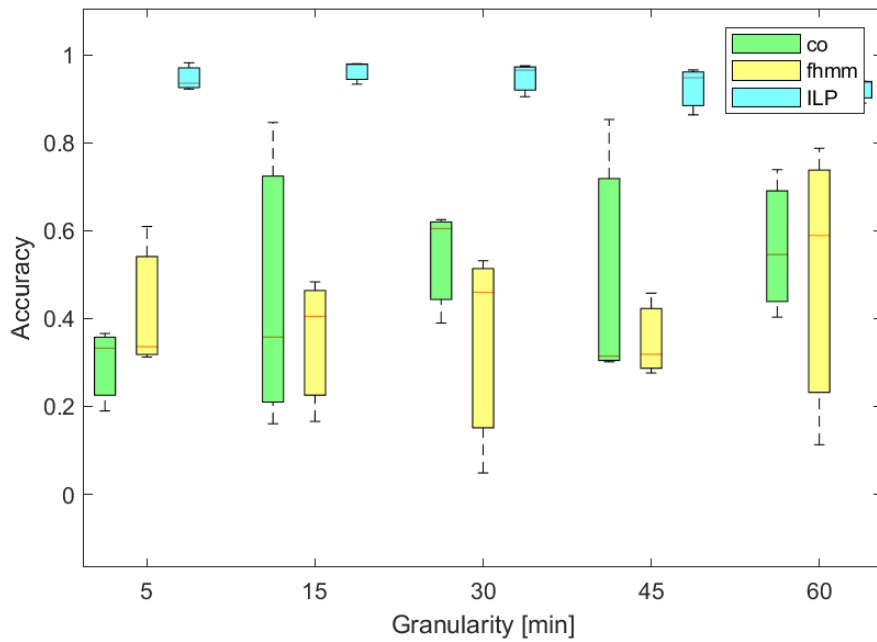


Figure 59: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

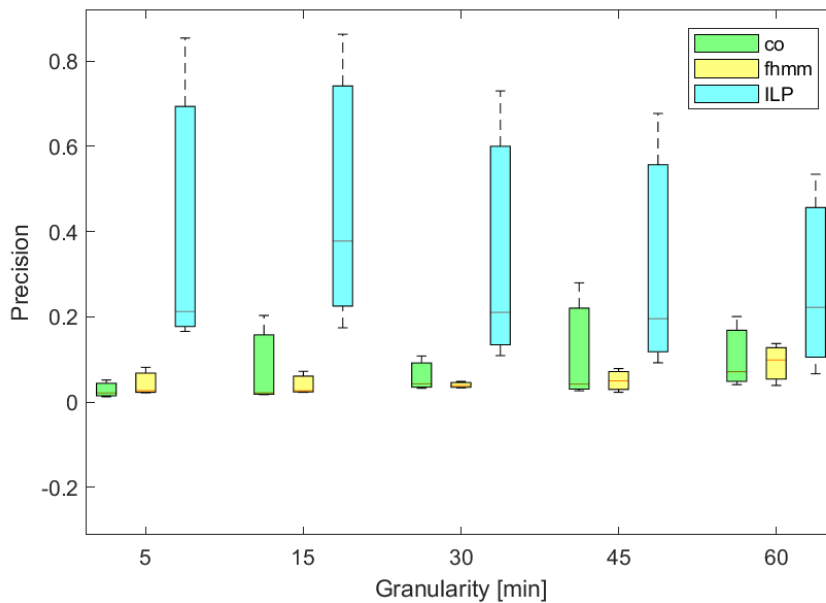


Figure 60: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating dishwasher in the UKDale dataset, for different measurement granularities.

Focusing on the disaggregation of dishwasher consumption patterns, results reported from Figure 55 to Figure 60 show that the ILP approach consistently outperforms the CO and FHMM algorithms in terms of normalized error, root mean square error, precision and accuracy. This is mainly due to the extremely high false positive rate exhibited by CO and FHMM, whereas the false negative rate achieved by ILP algorithm never surpasses 0.2.

4.4.2.2 Washer-Dryer Disaggregation

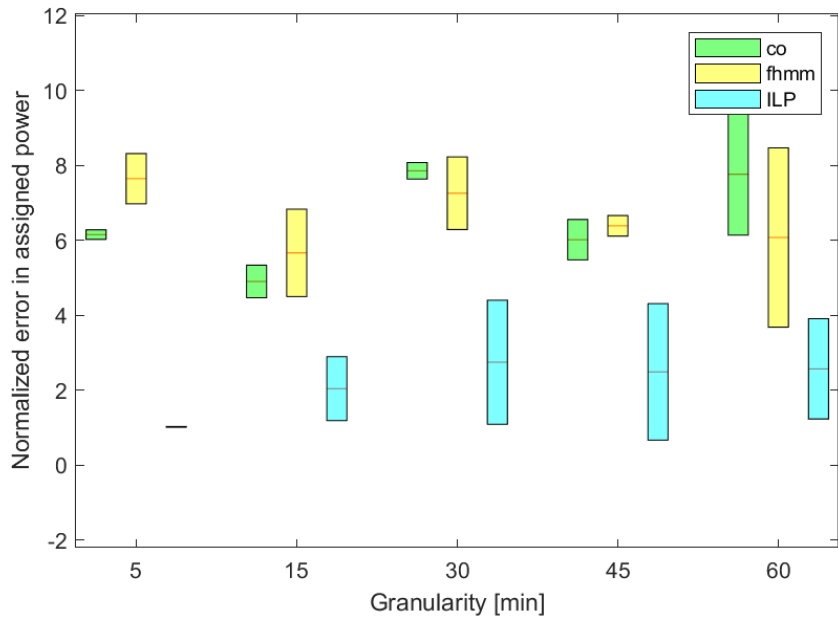


Figure 61: Normalized Error in Assigned Energy achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

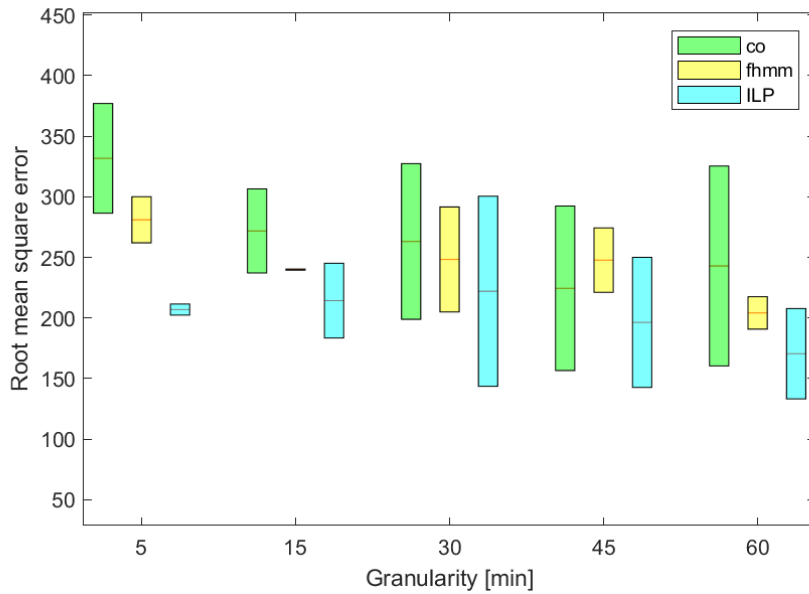


Figure 62: Root Mean Square Error achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

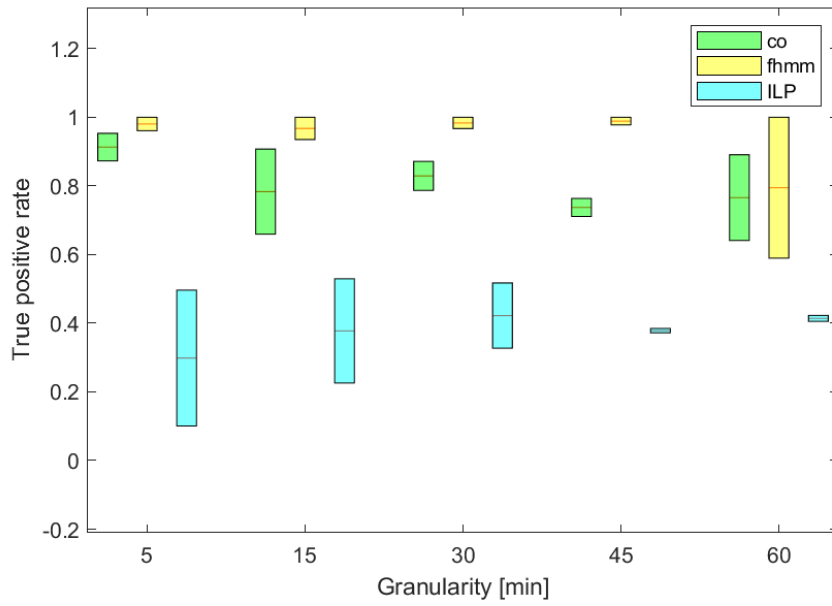


Figure 63: True Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

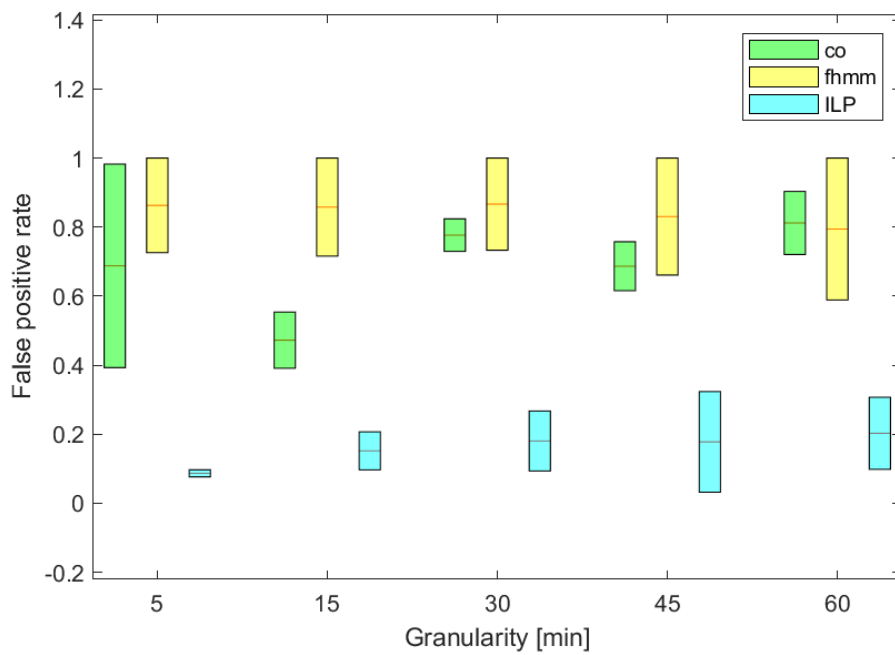


Figure 64: False Positive Rate achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

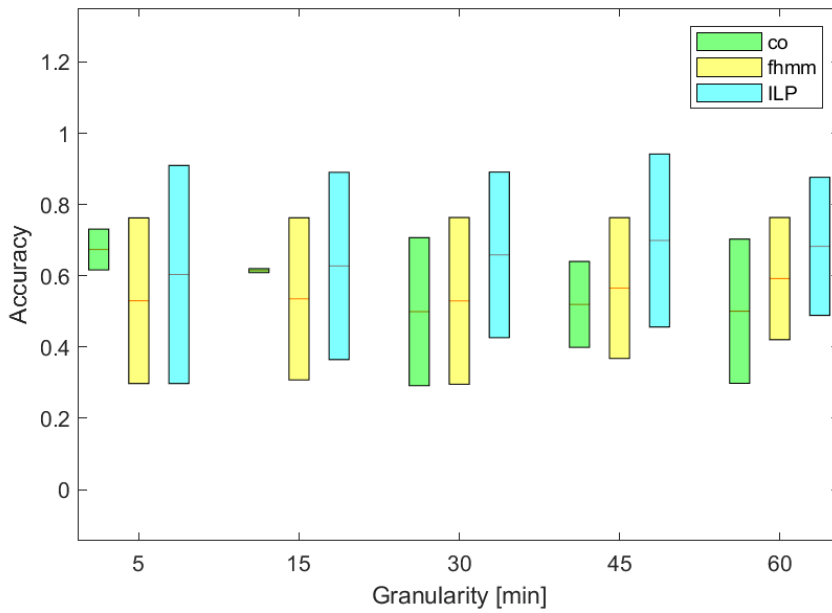


Figure 65: Accuracy achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

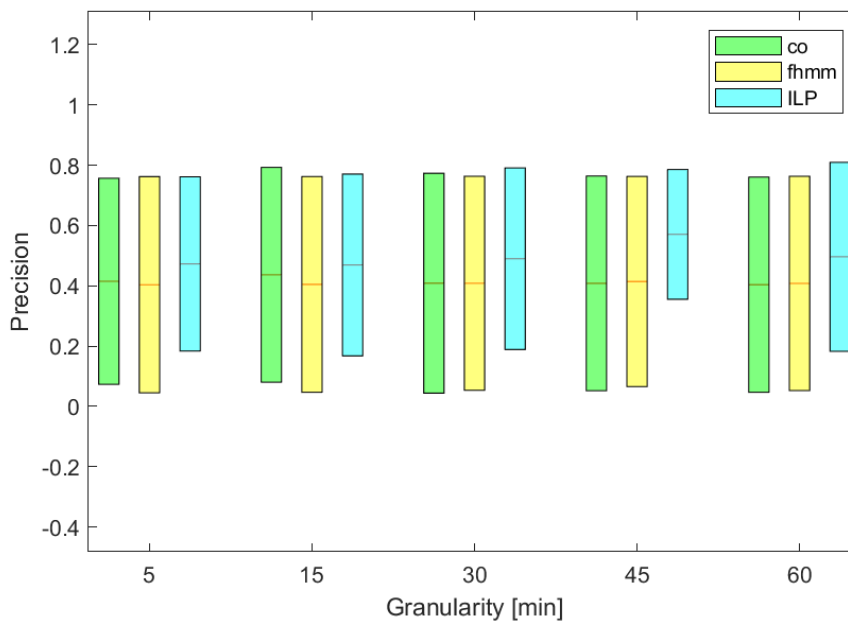


Figure 66: Precision achieved by the CO, FHMM and ILP algorithms when disaggregating washer-dryer in the UKDale dataset, for different measurement granularities.

Finally, when disaggregating the consumption patterns of the washer dryer, in the plots reported from Figure 61 to Figure 66, precision and accuracy values of the three algorithms appear to be comparable, with the ILP algorithm providing lowest normalized error and mean square error at all granularities, as well as lowest true and false positive rates, with respect to the two benchmarks. It is worth mentioning that the false positive rate of CO and FHMM is mostly above 0.8, whereas the ILP remains below 0.3 even in the case of 60 minutes granularity.

4.4.3 Assessment on SES user data

Finally, we report results obtained by the ILP algorithm when disaggregating the energy consumption patterns of 80 SES users participating to the enCOMPASS pilot.

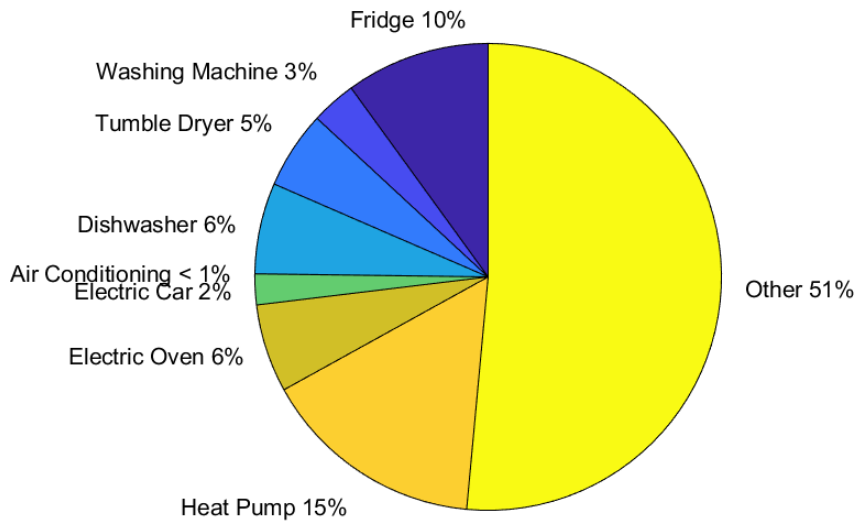


Figure 67: Average disaggregated consumption obtained with the ILP algorithm when disaggregating the SES dataset, at 15 mins granularity.

As no ground truth exists for such consumption profiles, the metrics reported in Section 4.3 cannot be computed. However, we can provide an estimation of the expected range of normalized error, root mean square error, fraction of correctly assigned energy, accuracy and precision as a function of the estimated amount of energy consumed by the appliances that do not belong to the set of devices being disaggregated. Intuitively, the higher the consumption of the appliances excluded by such set, the more difficult the disaggregation task is, since such fraction of consumed energy operates as noise that hides the consumption patterns of the appliances to be disaggregated.

To do so, we first consider the aggregate consumption collected from SES users (measurements were collected at 15 minutes granularities): on daily basis, for each aggregated consumption pattern of every user, we run the ILP algorithm implemented within the enCOMPASS platform, compute the percentage of consumed energy not attributed by the algorithm to appliances belonging to the predefined disaggregation set and plot the number of collected daily instances as a function of such percentage. Then, we consider the buildings belonging to the synthetic dataset described in Section 4.1 and plot the normalized error, root mean square error, fraction of correctly assigned energy, accuracy and precision (averaged over one month) versus the estimated percentage of energy attributed to appliances not belonging to the disaggregation set. Finally, we identify the region of space where most of the points are clustered, thus retrieving for every metric: *i*) the range of values where the majority of points lie; *ii*) the associated range of consumption percentage by non-disaggregated appliances. Results are reported from Figure 68 to Figure 72. It is worth noticing that accuracy values vary between 0.72 and 0.93 for users with percentage of energy consumption by non-disaggregated appliances between 30% and 90% (which cover the vast majority of SES users).

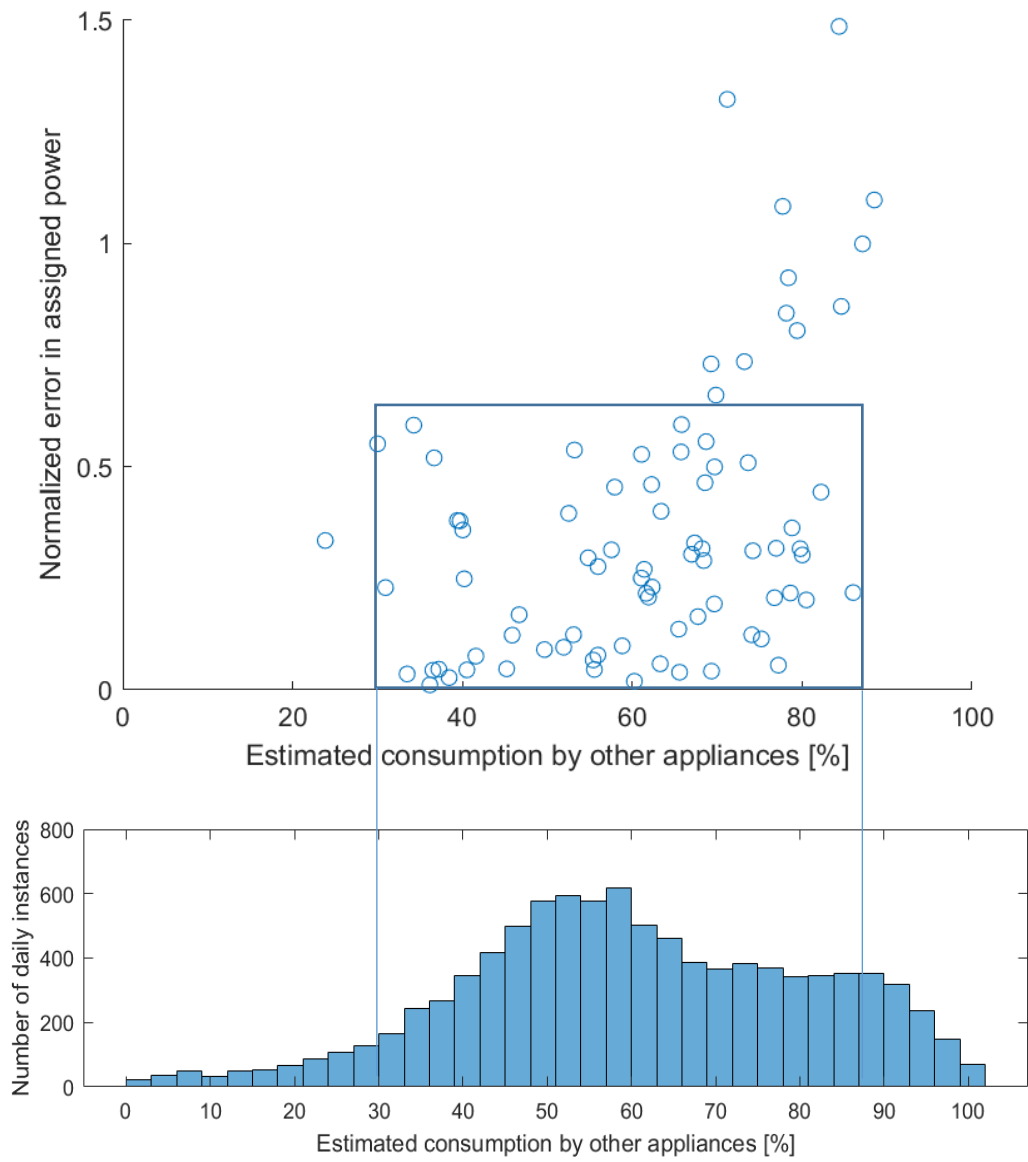


Figure 68: Estimated range of normalized error of assigned power.

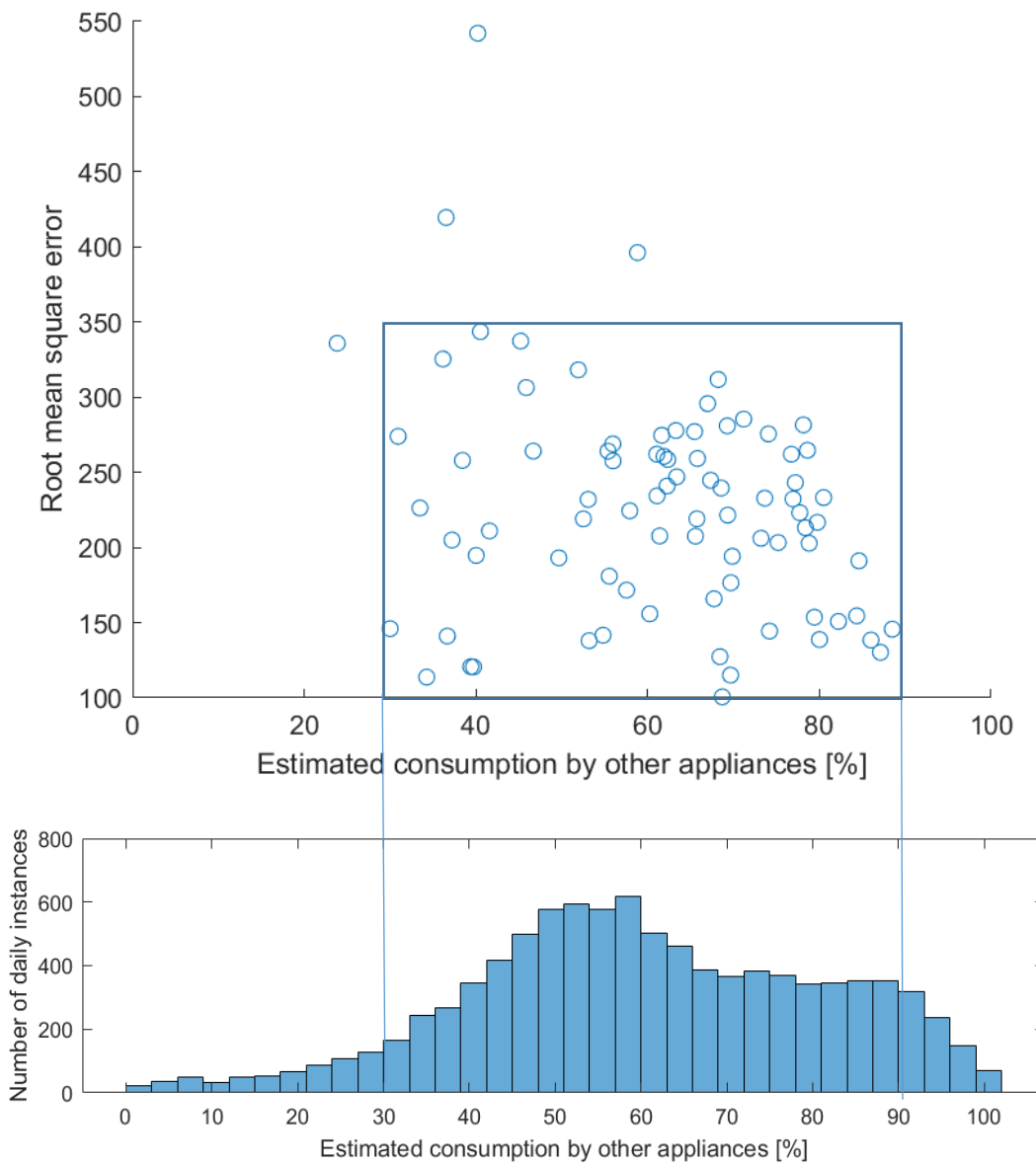


Figure 69: Estimated range of root mean square error.

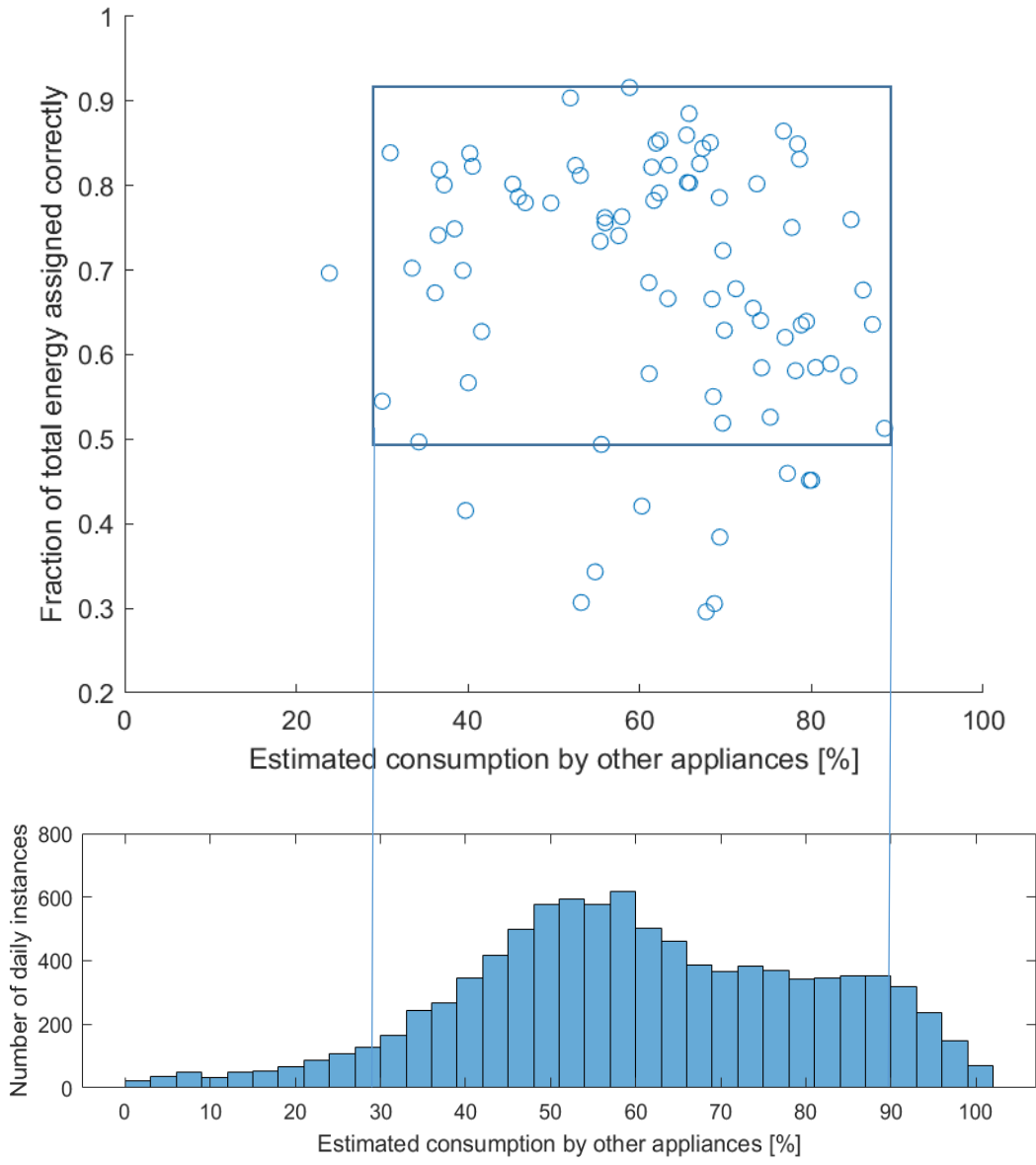


Figure 70: Estimated range of fraction of energy assigned correctly.

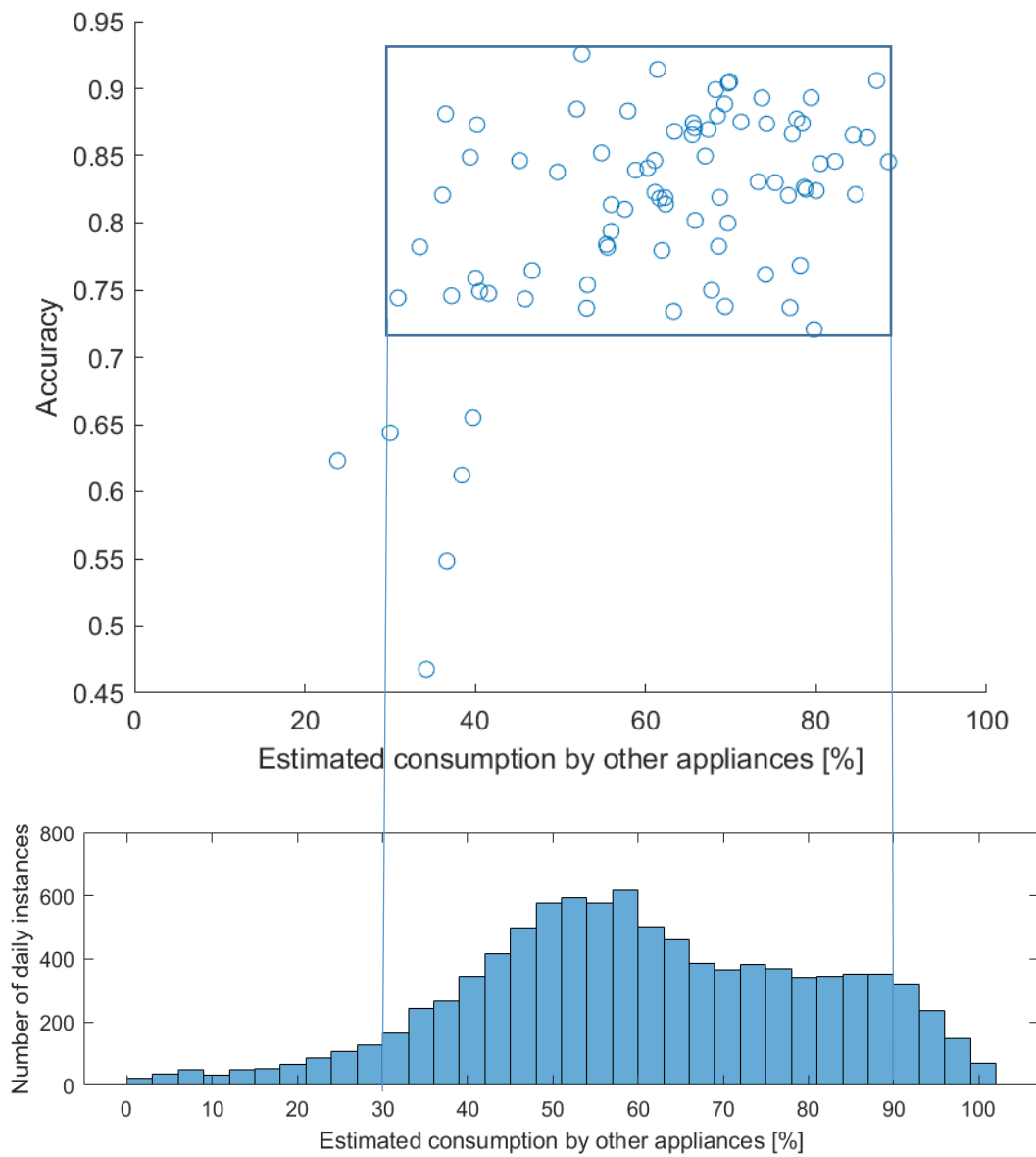


Figure 71: Estimated range of accuracy.

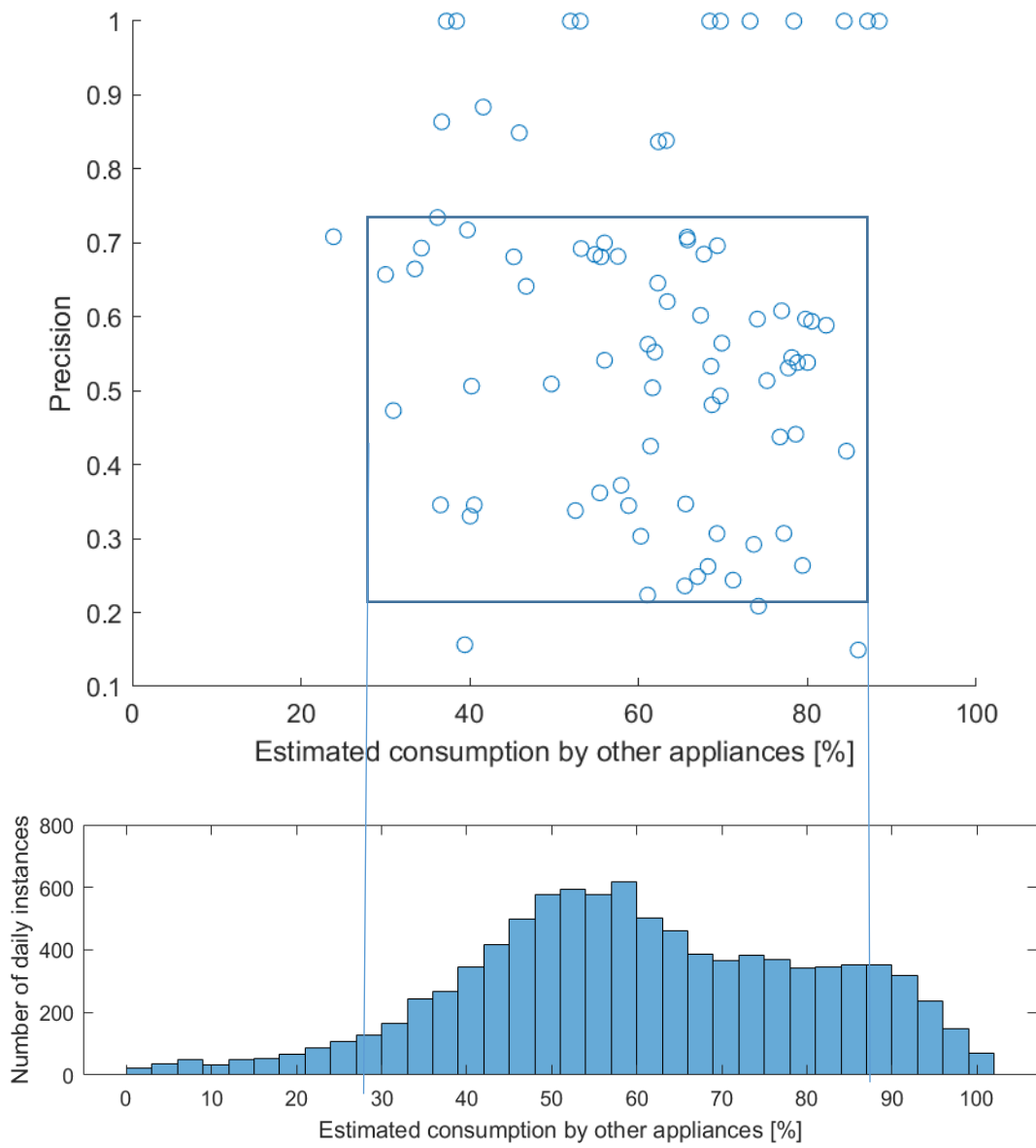


Figure 72: Estimated range of precision.

5 COMFORT INFERENCE ENGINE

Comfort could be described as all the things that contribute to a person’s physical ease and well-being. One of the most essential factors for a building is in what extent it is providing an environment that is comfortable for the occupants. In particular, comfort is how the occupants realize the indoor environmental conditions according to their needs and their objective senses. The main comfort values that an occupant can realize are strongly related to visual and thermal conditions in the building. In other words, thermal comfort is the sense of an occupants related to the indoor temperature, which is also affected by other factors, such as humidity, clothing, activity, etc. On the other hand, visual comfort is how an occupant realizes the luminosity in a room/ space, which is strongly affected by the external luminosity conditions, as well as the type of the lighting sources in the rooms. In this Section, the comfort inference engine aims to estimate the thermal and visual comfort in an indoor environment utilizing the indoor environmental conditions (temperature, humidity, luminosity), and the feedback from the user.

This section describes the way that Comfort Inference Algorithm is implied. The overall procedure is described in Figure 73. In Section 5.1, is described the thermal comfort estimation algorithm while in Section 5.2 the visual comfort approach. Finally, in Section 5.3 results from comfort inference algorithms in real-life environments are presented.

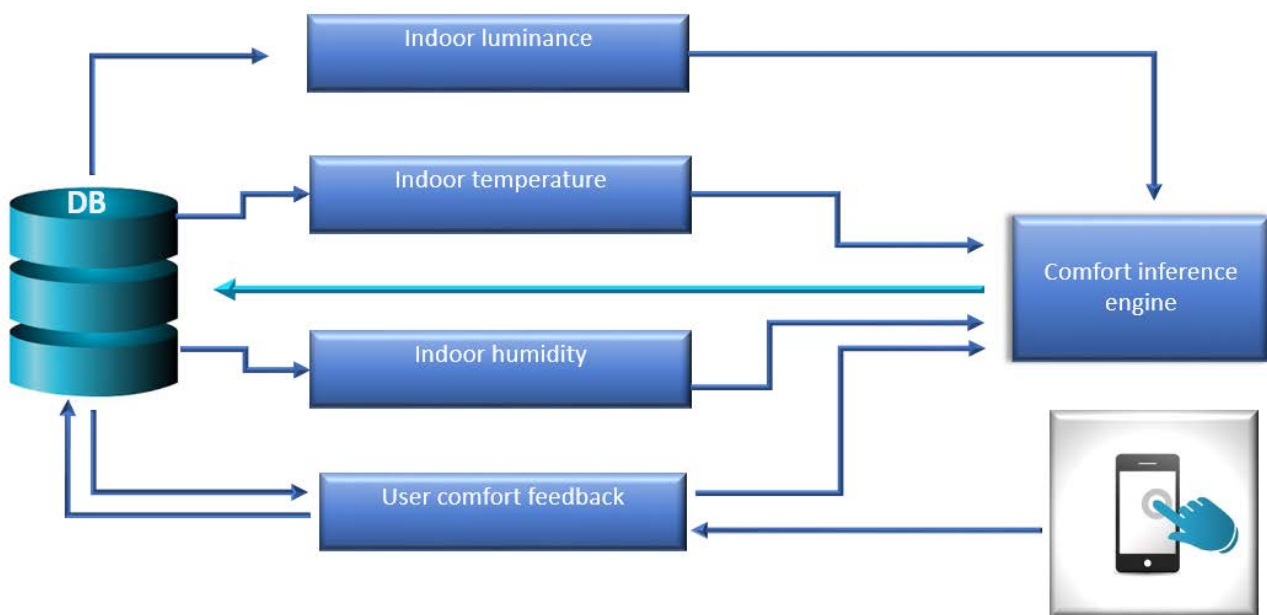


Figure 73: Comfort Inference Engine conceptual architecture.

5.1 THERMAL COMFORT

Thermal comfort is the state of mind which expresses satisfaction with the thermal conditions in an environment. A major objective for the indoor environment is to be convenient for the occupants, so thermal comfort estimation is a fundamental issue. Thermal comfort is calculated from the static PMV/PPD model developed by P.O. Fanger (Fanger 01). The parameters that are taken into consideration by the “Comfort Equation” are (Fanger 01):

- air temperature;
- mean radiant temperature;

- relative air velocity;
- humidity;
- activity level;
- clothing.

Since achieving optimal thermal comfort conditions is not always feasible, a level of discomfort psycho-physical scale has been defined by ANSI/ASHRAE Standard 55 (ASHRAE 03), calculating Predicted Mean Vote (PMV) (Olesen02) to a scale ranging from -3 to +3. In this scale, 0 is neutral, ±1 is slightly warm/cool, ±2 is warm/cool and ±3 is hot/cold, as shown in Figure 74.

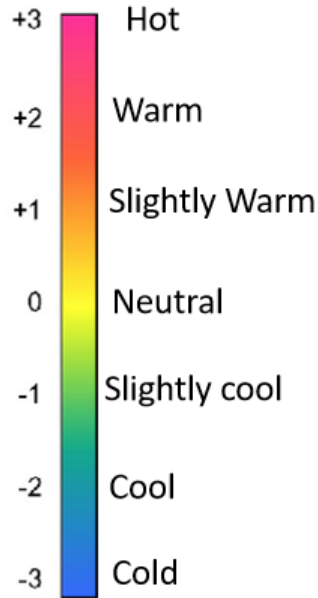


Figure 74: Thermal Comfort scale

The energy balance of the human body requires that the heat produced by the body is equal to the heat lost to the environment, so that the internal body temperature is kept approximately steady to 37°C, without being affected from variations in ambient temperature. In a steady-state heat balance, the heat energy produced by the metabolism equals the rate of heat transferred from the body by conduction, convection, radiation, evaporation and respiration. The fundamental thermodynamic process in heat exchange between man and his environment may be described by the general heat balance equation (Krinidis 14):

$$M - W = C + R + E_{sk} + (C_{res} + E_{res}). \quad (1)$$

The external work w (W/m^2) in the equation is small and is generally ignored under most situations. The internal energy production M (W/m^2) is determined by metabolic activity. C (W/m^2) is the heat loss by convection. R (W/m^2) is the heat loss by thermal radiation. E_{sk} (W/m^2) is the heat loss by evaporation from the skin. C_{res} (W/m^2) and E_{res} (W/m^2) are the sensible and the evaporation heat loss due to respiration respectively.

The convection heat transfer C (W/m^2) from the human body to the environment is given by:

$$C = f_{cl} \cdot h_c (T_{cl} - T_a), \quad (2)$$

where, T_{cl} (°C) is the clothing surface temperature and T_a (°C) is the ambient air temperature. The heat transfer coefficient h_c (W/m²K) depends on the air velocity V_a (m/s) across the body and consequently also upon the position of the person and orientation to the air current. An approximate value of h_c during forced convection can be evaluated as:

$$h_c = 12.1 \cdot V_a^{0.5} . \quad (3)$$

The clothing area factor f_{cl} is given by the following formulation:

$$f_{cl} = 1.05 + 0.1 \cdot I_{cl} , \quad (4)$$

where I_{cl} is the thermal insulation of clothing. The insulation of clothing is often expressed in clo units, but it is used in SI units as R_{cl} in calculations:

$$R_{cl} = 0.155 \cdot I_{cl} . \quad (5)$$

The rate of heat transfer by radiation R (W/m²) depends on the mean temperature of surrounding surfaces, skin or clothing surface temperature and properties of clothing (or skin) and surrounding surfaces. The radiation heat transfer between the body and surrounding surfaces is given as follows:

$$R = \sigma \cdot \varepsilon_{cl} \cdot f_{cl} \cdot F_{vf} \cdot [(T_{cl} + 273.15)^4 - (T_r + 273.15)^4] , \quad (6)$$

where ε_{cl} is the emissivity of the clothing. The emissivity of the clothing and skin is very close to that of a black body and thus has a value of nearly 1. F_{vf} is the view factor between the body and the surrounding surface, which determines the effective area of the body for radiation, and consequently is less than the total surface area (usually about 75% of the total). σ is the Stefan-Boltzmann constant, which is equal to $\sigma = 5.67 \times 10^{-8}$ W/m²K⁴. T_r (°C) and expressed the radiant temperature. The surrounding surface temperature is usually at a low temperature level. Thus, the temperature of the surrounding surfaces can be assumed that are equal to the ambient air temperature T_a (°C).

The respiration heat loss is divided into evaporative heat loss (latent heat) and sensible heat loss. The rate of the heat transfer by respiration is usually at the lower level beside the other rates of the heat transfer. This rate is given by:

$$RH = P_a / P_s , \quad (8)$$

where the RH is the relative humidity of the ambient air and P_s (P_{α}) is the saturation pressure of water vapor. The saturation pressure of water vapor is calculated as follows:

$$P_s = \left[0.782 + 2.692 \cdot \frac{T_a}{100} + 6.290 \left(\frac{T_a}{100} \right)^{2.325} \right]^2 \quad (9)$$

The rate of the heat loss by evaporation is the removal of heat from the body by evaporation of perspiration from the skin. Evaporation always constitutes a rejection of heat from body. The evaporation loss is dependent upon the mass transfer coefficient and the air humidity ratio for a given body surface temperature. The heat loss by evaporation is made up of two, the insensible heat loss by skin diffusion and the heat loss by regulatory sweating. This rate can be calculated by:

$$E_{sk} = 3.05 \cdot (5.73 - 0.007 \cdot M - P_a) + 0.42 \cdot (M - 58.15). \quad (10)$$

The conduction heat transfer through the clothing takes place in order to the different temperatures of its inner and outer surfaces. The conduction heat transfer from the inner surface to the outer surface of the clothing is defined as follows:

$$C_k = (T_{sk} - T_{cl}) / R_{cl}, \quad (11)$$

where T_{sk} (°C) is the skin temperature. This heat transfer from the clothing's outer surface is further transferred to the environment by convection and radiation heat losses. Thus:

$$C_k = C + R. \quad (12)$$

As the heat energy flow through the clothing in the steady-state is determined by (12) the clothing temperature can be determined using (11) as presented below:

$$T_{cl} = T_{sk} - R_{cl} \cdot (C + R) \Rightarrow$$

$$T_{cl} = T_{sk} - R_{cl} \cdot \left\{ \begin{array}{l} f_{cl} \cdot h_c \cdot (T_{cl} - T_a) + \sigma \cdot \epsilon_{cl} \cdot f_{cl} \cdot F_{vf} \cdot \\ \left[(T_{cl} + 273.15)^4 - (T_r + 273.15)^4 \right] \end{array} \right\} \quad (13)$$

T_{sk} (°C) is defined as follows:

$$T_{sk} = 35.7 - 0.0275 \cdot M. \quad (14)$$

Finally, the PMV value is determined from the following equation (Fanger 01):

$$PMV = (0.303 \cdot e^{-0.036 \cdot M} + 0.028) \cdot L, \quad (15)$$

where L is defined as follows:

$$L = M - W - C - R - E_{sk} - (C_{res} + E_{res}) \quad (16)$$

All, the above equations are in accordance with Fanger (Fanger 01). Consequently, in order to compute thermal comfort four factors must be taken into consideration: indoor temperature (T_a), Indoor humidity (RH), Clothing (I_{cl}) and activity (M), as follows:

$$PMV(T_a, RH, I_{cl}, M) \quad (17)$$

The indoor temperature (T_a) and indoor humidity (RH) are measured by indoor sensors. The other two factors must be taken into consideration are the Clothing (I_{cl}) and the Metabolic rate (M). Clothing is the resistance to sensible heat transfer provided by a clothing ensemble and it is expressed in clo units. A clo unit is used to express the thermal insulation provided by garments and clothing ensembles, where $1 \text{ clo} = 0.155 \text{ m}^2 \cdot \text{°C/W}$ ($0.88 \text{ ft}^2 \cdot \text{h} \cdot \text{°F/Btu}$) (ASHRAE 03). Metabolic rate (M) is the rate of transformation of chemical energy into heat and mechanical work by metabolic activities within an organism, usually expressed in terms of unit area of the total body surface (ASHRAE 03).

The above personal factors cannot be calculated from a sensor, neither there is any feasible way to receive these from the occupants, especially in a daily basis, even worst every 15 minute. Consequently, an initial assumption was made based on ANSI/ASHRAE Standard 55 (ASHRAE 03). Thermal comfort is calculated for each user. However, when the user gives his/her comfort feedback, the I_{cl} and M parameters are predicted (Section 5.1.2). Afterwards, the thermal comfort is calculated with the predicted clothing and activity values. The overall procedure is described in Figure 75.

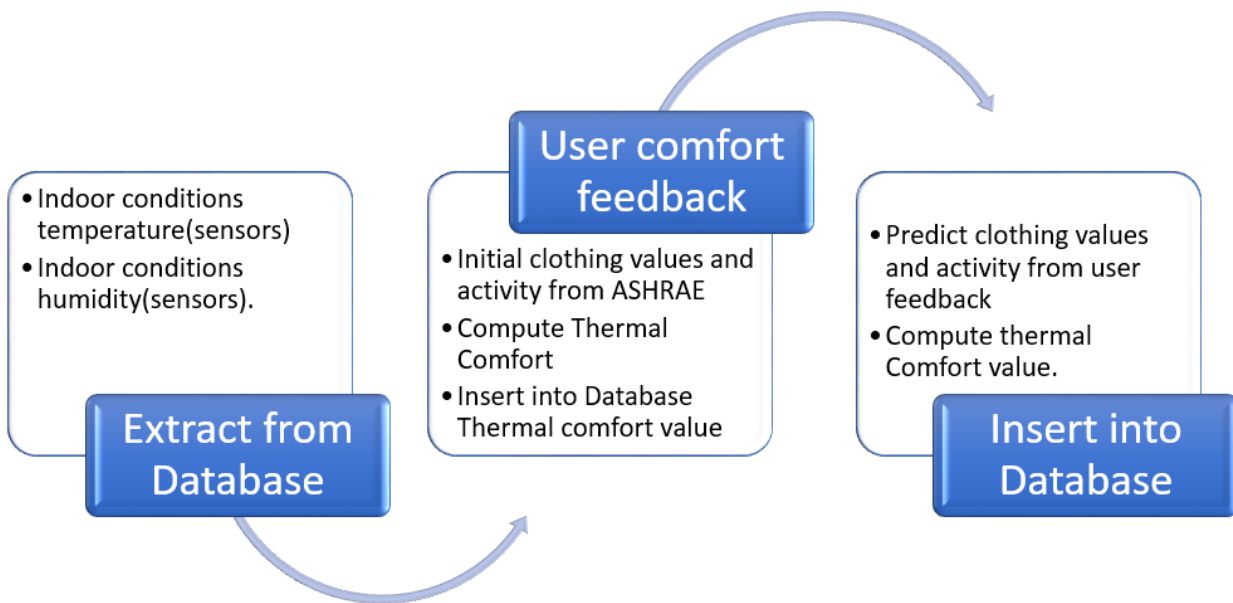


Figure 75: Computation of Thermal comfort

5.1.1 Sensor Data preprocessing

Data preprocessing is an essential step for the Thermal comfort inference, since the quality of the data affects significantly the outcome. Before feeding data into the inference engine, all the needed information and data are preprocessed.

- Duplicated values. Data frequency is 15 minute, thus the input values are checked for duplications by comparing their insertion timestamps. In case of duplication, two policies are followed:
 - If the duplicated values are equal, then one of them is kept.
 - If the values are not equal, average value is utilized.
- Null or incomplete values. When a value is set to null or does not exist, based on the frequency of the data, then it is filled using linear interpolation since both temperature and humidity are both time series data with constant step. In order to evaluate it to our dataset, data from the SES pilot, were randomly deleted and the results of the interpolation were compared with the real values. As it is shown in Figure 76, the error is insignificant.

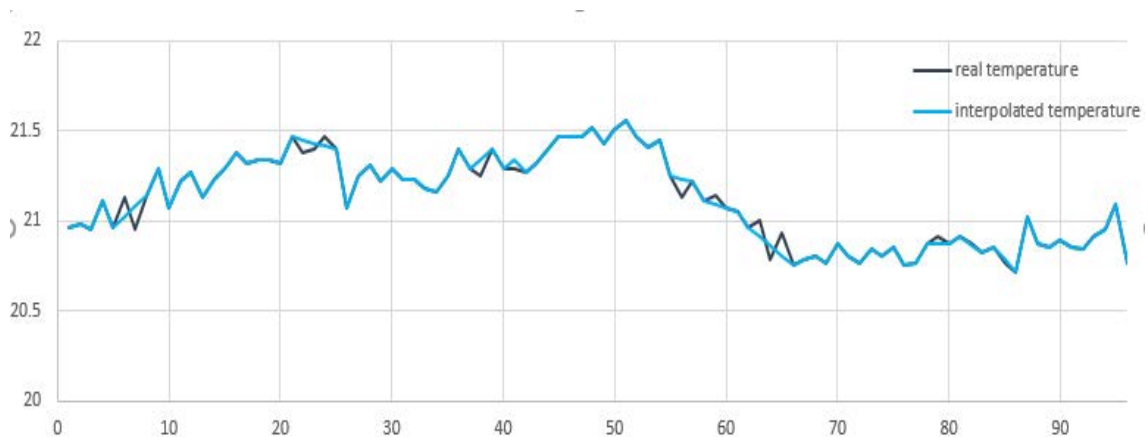


Figure 76: Example of temperature interpolation

5.1.2 User Feedback and prediction of Metabolic Rate and Activity

As long as the users provide no feedback, their comfort is calculated from sensor data (T_a , RH) acquired from the pilots and the metabolic rate (M) and clothing insulation (I_{cl}) data initialized according to ASHRAE standard (ASHRAE 03). These values are shown in Table 2 and Table 3. The hypothesis for the metabolic rate and clothing insulation values begins by separating the day at five intervals. Regarding the metabolic rate, mild activities such as sitting, reclining, typing, reading are considered for each user during the day, while at night the user is considered to be sleeping. The clothing insulation takes into the consideration the day interval in combination with the season of the year, in order to infer a user's clothing. Summer clothing values for example, are much lower than winter's. Day and night separation is also made here, since during sleep the bed and the sheets provide some extra insulation.

Table 2: Metabolic rates for typical tasks (ASHRAE 03)

TABLE A1 Metabolic Rates for Typical Tasks

Activity	Metabolic Rate		
	Met Units	W/m ²	(Btu/h-ft ²)
Resting			
Sleeping	0.7	40	(13)
Reclining	0.8	45	(15)
Seated, quiet	1.0	60	(18)
Standing, relaxed	1.2	70	(22)
Office Activities			
Reading, seated	1.0	55	(18)
Writing	1.0	60	(18)
Typing	1.1	65	(20)
Filing, seated	1.2	70	(22)
Filing, standing	1.4	80	(26)
Walking about	1.7	100	(31)
Lifting/packing	2.1	120	(39)

Table 3: Garment Insulation (ASHRAE 03).

TABLE B2
Garment Insulation^a

Garment Description [†]	I_{cl} , clo	Garment Description ^b	I_{cl} , clo
Underwear		Dress and Skirts^{**}	
Bra	0.01	Skirt (thin)	0.14
Panties	0.03	Skirt (thick)	0.23
Men's briefs	0.04	Sleeveless, scoop neck (thin)	0.23
T-shirt	0.08	Sleeveless, scoop neck (thick), i.e., jumper	0.27
Half-slip	0.14	Short-sleeve shirtdress (thin)	0.29
Long underwear bottoms	0.15	Long-sleeve shirtdress (thin)	0.33
Full slip	0.16	Long-sleeve shirtdress (thick)	0.47
Long underwear top	0.20	Sweaters	
Footwear		Sleeveless vest (thin)	0.13
Ankle-length athletic socks	0.02	Sleeveless vest (thick)	0.22
Pantyhose/stockings	0.02	Long-sleeve (thin)	0.25
Sandals/thongs	0.02	Long-sleeve (thick)	0.36
Shoes	0.02	Suit Jackets and Vests^{††}	
Slippers (quilted, pile lined)	0.03	Sleeveless vest (thin)	0.10
Calf-length socks	0.03	Sleeveless vest (thick)	0.17
Knee socks (thick)	0.06	Single-breasted (thin)	0.36
Boots	0.10	Single-breasted (thick)	0.44
Shirts and Blouses		Double-breasted (thin)	0.42
Sleeveless/scoop-neck blouse	0.12	Double-breasted (thick)	0.48
Short-sleeve knit sport shirt	0.17	Sleepwear and Robes	
Short-sleeve dress shirt	0.19	Sleeveless short gown (thin)	0.18
Long-sleeve dress shirt	0.25	Sleeveless long gown (thin)	0.20
Long-sleeve flannel shirt	0.34	Short-sleeve hospital gown	0.31
Long-sleeve sweatshirt	0.34	Short-sleeve short robe (thin)	0.34
Trousers and Coveralls		Short-sleeve pajamas (thin)	0.42
Short shorts	0.06	Long-sleeve long gown (thick)	0.46
Walking shorts	0.08	Long-sleeve short wrap robe (thick)	0.48
Straight trousers (thin)	0.15	Long-sleeve pajamas (thick)	0.57
Straight trousers (thick)	0.24	Long-sleeve long wrap robe (thick)	0.69
Sweatpants	0.28		
Overalls	0.30		
Coveralls	0.49		

User feedback is utilized in order to revise the initial metabolic rate and clothing insulation values to converge to the user's objectives. The final goal is to **personalize thermal comfort inference for each user**, since every person may have different dressing preferences and may perform different activities in his/her house during the day. The feedback provided by enCOMPASS users refers to their thermal sensation in terms of the PMV index. The correction of the metabolic rate and clothing insulation values is made only for the specific interval that the feedback is given. The new M , I_{cl} are calculated according to the following methodology.

The first step towards building this model was the formulation of a training dataset. According to Ashrae standards (ASHRAE 03), indoor activity, clothing insulation and metabolic rate have upper and lower limits. Discrete values were chosen within the respective boundaries for all of the variables that compose Fanger's equation (T_{a} , RH , I_{cl} , M). The step that was used for the sampling of each variable, was selected considering the variable's impact on the final PMV outcome at the [-3, 3] scale of PMV (Table 4).

Table 4: Variables sampling for the training dataset

Variable	Interval	Step	Impact of variable's step change on PMV value
Temperature	[18, 31]	0.3	~0.1
Humidity	[20, 75]	5	~0.05
Metabolic Rate	[44, 164]	4	0.05 - 0.2
Clothing Insulation	[0.04, 1.84]	0.04	0.02 - 0.2

The next step requires solving Fanger's equation for all of the possible states that were generated. The combination of T_a , RH , I_{cl} , M values generate a total of 770.400 different states. After solving the equation, a mapping table was formulated that will be utilized as the training dataset for our model (Table 5).

Table 5: Mapping table

Temperature	Humidity	Clothing Insulation	Metabolic Rate	PMV
23.4	40	0.84	44	-0.973834205
23.4	40	0.84	48	-0.762835508
23.4	40	0.84	52	-0.58211906
23.4	40	0.84	56	-0.412545543
23.4	40	0.84	60	-0.246522715
23.4	40	0.84	64	-0.877548226
23.4	40	0.84	68	0.0989686
23.4	40	0.84	72	0.2179506
23.4	40	0.84	76	0.322736486
23.4	40	0.84	80	0.416239678
23.4	40	0.84	84	0.500847436
23.4	40	0.84	88	0.578509586
23.4	40	0.84	92	0.65081292
23.4	40	0.84	96	0.719043497
23.4	40	0.84	100	0.784238774
23.4	40	0.84	104	0.847231152
23.4	40	0.84	108	0.908684324
23.4	40	0.84	112	0.969123551
23.4	40	0.84	116	1.028960859
23.4	40	0.84	120	1.088515965
23.4	40	0.84	124	1.148033625
23.4	40	0.84	128	1.207697993
23.4	40	0.84	132	1.26764447
23.4	40	0.84	136	1.327969474
23.4	40	0.84	140	1.388738457
23.4	40	0.84	144	1.449992479
23.4	40	0.84	148	1.511753575
23.4	40	0.84	152	1.574029117
23.4	40	0.84	156	1.63681535
23.4	40	0.84	160	1.700100249
23.4	40	0.84	164	1.763865801

When an enCOMPASS user decides to give feedback about the thermal comfort conditions in his/her house, the given value is considered to be the actual PMV value for the specific timestamp. The task of our model is to use the given feedback along with the sensor data and predict the clothing insulation and the metabolic

rate. The formulated problem requires the estimation of multiple continuous variables $y_i=(M, I_{cl})$ from a vector of k input variables $x_i=(PMV_feedback, T, H)$. This is a multi-target regression (MTR) problem so an appropriate regressor is selected. To this end, extremely randomized trees (extra trees), presented by Geurts et al (Geurts 04) was utilized. Extra trees is an algorithm for ensemble tree construction based on extreme randomization. It belongs to the global methods of MTR, which means that all of the target variables are predicted simultaneously using one model in contrast to the local methods that predict each target variable separately. Global methods exploit the dependencies that exist between the target variables and result in better predictive performance.

The extra trees regression algorithm builds an ensemble of unpruned regression trees according to the classical top-down procedure. It has two main differences with other ensemble tree-based methods:

- The procedure of selecting cut-points for splitting the tree nodes is performed randomly.
- The trees grow using the whole learning sample and not just a bootstrap replica.

The splitting procedure for numerical attributes includes the following parameters:

- K , which denotes the number of attributes selected at each node;
- n_{min} , which refers to the minimum sample size for splitting a node;
- L , which represents the number of trees of the ensemble.

The final prediction in regression problems is given by aggregating the predictions of all trees and then using the arithmetic average. From variance point of view, extra trees are able to reduce variance more strongly than other randomization tree methods, using explicit randomization of the cut-point and attribute, combined with ensemble averaging. Bias is also minimized by the usage of the full original learning sample, in contrast to methods that use bootstrap replicas. Assuming balanced trees, the complexity of tree growing is of order $N \cdot \log N$ with respect to learning sample size. This is common for the most tree growing procedures, but the simplicity of the node splitting process in extra trees results in a much smaller constant factor comparing to other methods which locally optimize cut-points. The parameters K , n_{min} , L can be adjusted manually or automatically, however it is suggested by Geurts that the default settings are used in order to maximize the computational advantages and autonomy of the method. The above claim is empirically confirmed at our case, since different settings of the algorithms were used, but finally the default setting were selected as they provided more accurate results. The default criteria for measuring the quality of a split is mean squared error.

The users are able to give their feedback through the enCOMPASS mobile app (Figure 77). Then, the model uses the feedback along with temperature and humidity data and finally predicts the new values of metabolic rate and clothing insulation for the current user. Those values refer to the time interval in which the feedback is given. Metabolic rate and clothing insulation values are finally stored, updating previous ones. From that point on, the user's thermal comfort will be estimated with these new M and I_{cl} values.



Figure 77: Mobile app feedback screenshot.

Summarizing, the comfort inference algorithm is executed as a whole, executing the following steps (Figure 78):

- All the necessary data are retrieved from the database. This includes: temperature, humidity, metabolic rate, clothing insulation, consumption.
- Data are pre-processed in order to handle abnormalities such as null or duplicate values.
- Thermal comfort is being calculated using Fanger's equation.
- It is checked whether the user has given feedback regarding thermal comfort.
 - If not, calculated thermal comfort values are stored on the database.
 - If there is feedback, then the comfort feedback predictive model is loaded and new values for M and I_{cl} are calculated. Then, those values update the older ones on the database and the thermal comfort values are stored as well.

Finally, Figure 78 depicts the main steps of the thermal comfort inference algorithm developed within enCOMPASS project:

1. All the necessary data are retrieved from the database. This includes: temperature, humidity, metabolic rate, clothing insulation, consumption.

2. Data are pre-processed in order to handle abnormalities such as null or duplicate values.
3. It is checked whether the user has provided feedback. If there is feedback, then the comfort feedback predictive model is loaded and new values for M and I_{cl} are calculated.
4. Thermal comfort is being calculated using Fanger's Equation.

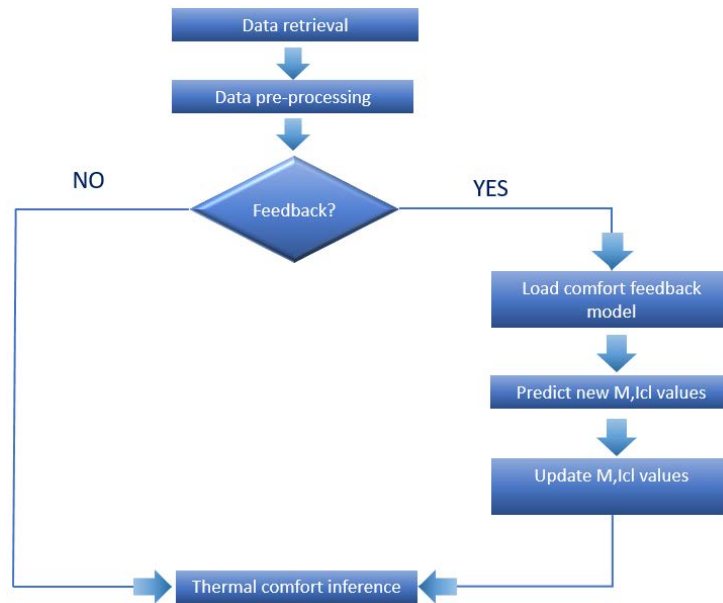


Figure 78: Thermal comfort flow chart

5.2 VISUAL COMFORT

Conversely to the specific meaning of thermal comfort, there is not a general meaning for visual comfort in the literature (Carlucci 09). As a consequent, many recent studies aim to estimate some specified aspects of visual comfort by characterizing the relationship between the human needs and the light environment, such as an available amount of light, light uniformity, light quality in rendering colours, and predicting the risk of glare for space occupants (Carlucci 09).

Calculation of the visual comfort turned out to be a challenging task, due to lack of data that are commonly used in the estimation of indoor comfort/discomfort with respect to the luminance conditions. The only available information from the enCOMPASS pilots is the luminance sensor. After reviewing the available visual comfort indices that could be used (Carlucci 09), it was decided to use Wienold's simplification of Discomfort Glare Probability. Discomfort Glare Probability (DGP) is a short-term, local, one-tailed index assessing glare, which was introduced in (Wienold 10) and validated in (Wienold 11) and given by:

$$DGP = 5.87 \cdot 10^{-5} E_v + 0.0918 \cdot \log_{10} \left[1 + \sum_{i=1}^n \left(\frac{L_{s,i}^2 \cdot \omega_{s,i}}{E_v^{1.87} \cdot P_i^2} \right) \right] + 0.16, \quad (17)$$

where E_v is the vertical eye illuminance, produced by the light source [lx]; L_s the luminance of the source (cd/m^2); ω_s the solid angle of the source seen by an observer; P is the position index, which expresses the change in experienced discomfort glare relative to the angular displacement of the source (azimuth and elevation) from the observer's line of sight. The equation is valid within the range of DGP between 0.2 and 0.8.

The above formula contains variables such as the solid angle and the position index, which are not available from the pilots. Thus, a simplification of the DGP is utilized, suggested by Hviid (Hviid 12) that just uses and depends linearly on the vertical illuminance at eye level:

$$DGPs|_{Hviid} = 5.87 \cdot 10^{-5} \cdot E_v + 0.16 \quad (18)$$

This simplification is similar to Wienold’s version of DGP, predicting a probability 2-3% lower than Wienold for the same value of E_v .

According to (Jakubiec 13), the values of DGP correspond to some degrees of glare sensation (Imperceptible, Perceptible, Disturbing, and Intolerable). In our case, similarly to the thermal comfort, a scale is defined from -3 to +3 in order to express the amount of indoor luminance. In this scale, -3 is complete dark and +3 is intolerable light, as shown in Figure 79.

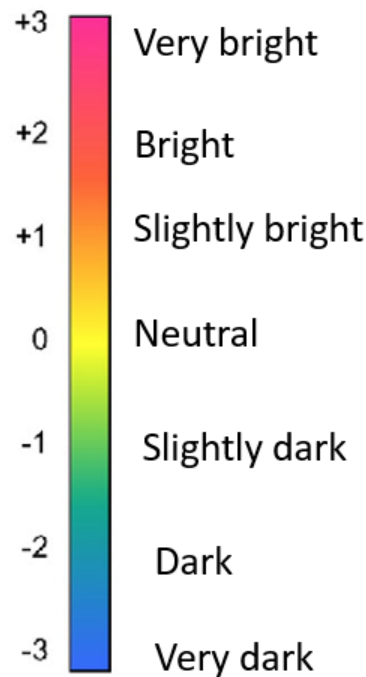


Figure 79: Visual Comfort scale

Hviid’s approach may have a relatively large margin of error because of the simplifications that are made. A second method is decided to be used along with Hviid’s equation aiming at achieving greater accuracy at visual comfort inference. The second method is based on the European Standard EN12464-1 referring to indoor lighting of work places (EN12464-1 14). The manual suggests that if the visual conditions differ from the normal assumptions, the luminance value may be adjusted by at least one step in the scale of illuminances presented in Figure 80.



Figure 80: Scale of illuminances (lux)

The above scale is fitted into a curve that corresponds to the [-3, 3] scale which is used for the description of visual comfort. Finally, both methods (Hviid referred as $DGPs|_{Hviid}$ and EN12464-1 referred as En) are used to the final calculation of the visual comfort. The final equation used to calculate the visual comfort is:

$$Visual = aDGPs|_{Hviid} + (1 - a)En \quad (19)$$

As described above, the visual comfort is calculated for each user. A flow chart of the algorithm is depicted in Figure 81.

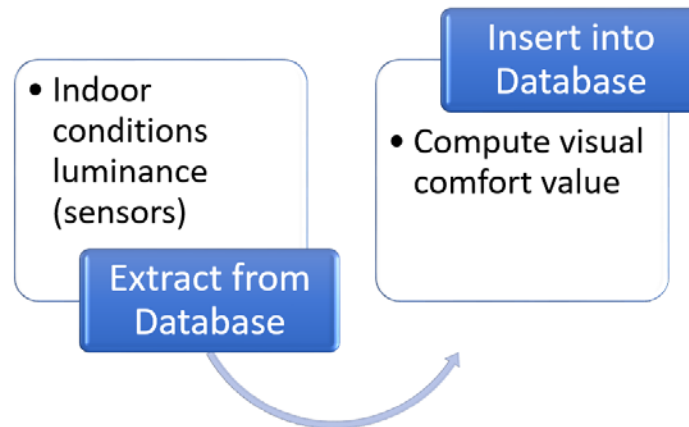


Figure 81: Compute of Visual comfort

5.2.1 Visual comfort Inference

Indoor lightning can be performed in three ways. Natural lightning or daylight, which is the part of solar radiation which is visible to the human eye. Artificial lightning which is the light produced by electrical means. And a combination of natural and artificial lightning which is the light produced by the daylight entering from windows and by electrical means. Moreover, artificial light is meant to either bring light where the natural light is not enough or to replace the natural night when night falls.

The visual comfort needs to be estimated under all daylight conditions, as the measurements are on a daily basis. The human perception of the luminous environment needs to have different metrics during the day. These metrics need to account the daylight variation as the light changes dynamically with time (Giovannini 15). During experiments at the CERTH smart house it was concluded that especially during the night the perception of “brightness” differentiated during dark hours. As a result, before implementing the visual inference model time intervals are checked if they are day or night. This discrimination is uniquely done for each pilot and per month, as the time it dawns and sets depends both from the month and the location. Two models were tested, one all-day model and one that distinguished the day from night. Both models use the equation (19). The all-day model uses the same coefficient a , during all day. On the other hand, the day-night model uses different coefficients for day and night. The difference between those models is depicted in Figure 82. The basic difference is that the all-day model estimates a lower visual comfort during dark hours. During the bright hours of the day, visual comfort remains the same for both models.

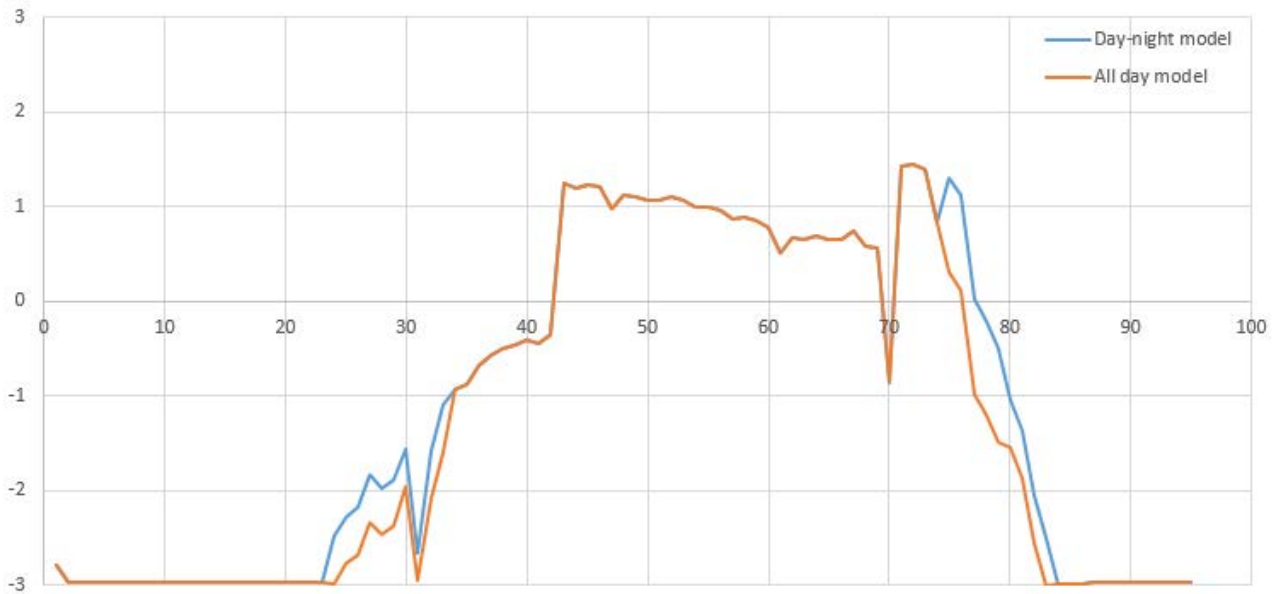


Figure 82: Test of night-day model and all day model from a SES user

Experimentally has been found that the visual comfort is estimated by equation (19), where coefficient α set to $\alpha=0.7$ during the night and $\alpha=0.6$ during the day.

5.2.2 Visual comfort Prediction

In the case of the Greek pilot (WVT), sensor data are only available for temperature and humidity. There are no luminance sensors installed and as a result the visual comfort can't be calculated as in the German (SHF) and the Swiss (SES) pilot. A predictive algorithm attempts to fill this gap. The algorithm is trained using data provided by SHF and SES pilot and after the appropriate feature selection, luminance and by extension visual comfort is predicted for WVT users.

All of the data that were available from the sensors and meters were initially tried out as features. Temperature, humidity and consumption did not provide sufficiently good results for the prediction of the target variable, namely luminance. Analyzing the database, it was noticed that luminance values were close to 0 at night, while they was a steep increase and decrease on the morning and the afternoon respectively. The values remained high during daytime sometimes reaching over 1000 lux (Figure 83).

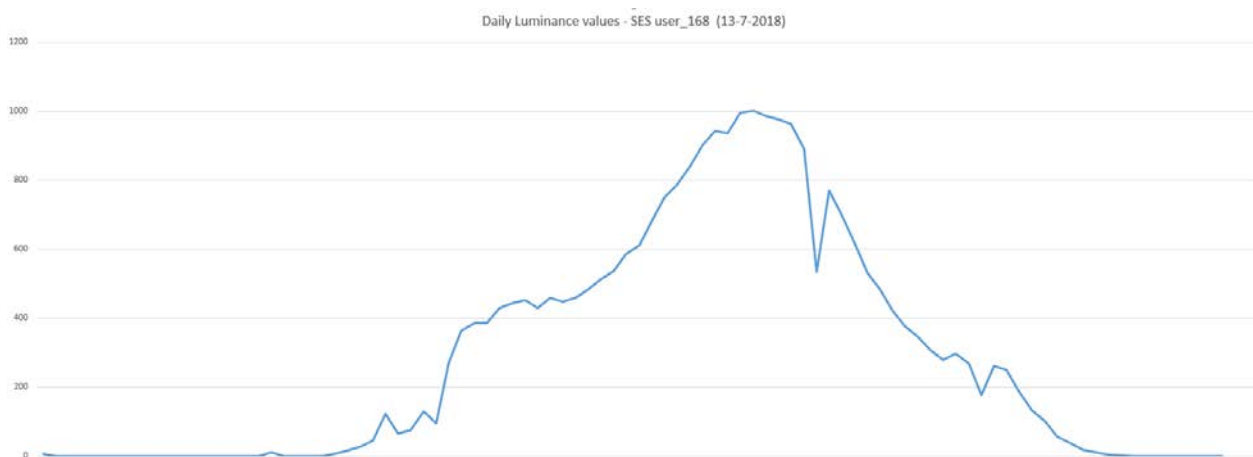


Figure 83: Daily luminance curve

It becomes obvious that the target variable is highly correlated with time. Hence, it was decided to formulate a feature that incorporates time dependency to the model. Using historical luminance data from SES and SHF database, the following clusters were formed:

Table 6 Time clustering of luminance data

Cluster number	Time Interval	Average luminance
1	00:00 – 06:00	2 lux
2	06:00 – 08:00	57 lux
3	08:00 – 09:00	131 lux
4	09:00 – 17:00	200 lux
3	17:00 – 18:00	132 lux
2	18:00 – 21:00	50 lux
1	21:00 – 24:00	4 lux

Table 6 confirms that luminance values remain very low during the night, so it was decided to create two separate models for the Greek pilot luminance prediction, night model and day model. Night model is used for the predictions that correspond to the time interval of cluster 1. Day model corresponds to clusters 2,3 and 4, so an additional feature that includes time dependence to the model was created.

The selected features for each model are presented below. Humidity data were removed as they had no contribution to the model’s performance.

Table 7 Features used at each model

Model	Features used
Day model	Temperature, Consumption, Time interval
Night model	Temperature, Consumption

Luminance prediction is a regression problem as the exact value of the luminance for each period needs to be predicted. Random forest regression is used for the training of the models. Random forests proposed by (Breiman 16) add an additional layer of randomness to bagging. In standard trees, each node is split using the best split among all variables while in random trees the best split among a subset of predictors randomly chosen at that node, is selected. This strategy performs very well and is robust to overfitting. The algorithm includes the following steps (Liaw 17):

- 1) Draw n_{tree} bootstrap samples from the original data.
- 2) Grow an unpruned tree for each of the samples. Chose the best split by randomly sampling the predictors.
- 3) Predict new data by aggregating the predictions of the n_{trees} . The average of the predictions is used for regression problems.

Algorithm parameters were tuned empirically, according to the results that were obtained with different configurations. The number of trees in the forest ($n_{estimators}$) were set to 100 and maximum depth of the tree (max_depth) was set to 12. The quality of the split is measured using mean squared error as a criterion. Minimum number of samples required to split an internal node ($min_samples_split$) is left to its default value.

Luminance prediction for the Greek pilot is part of the comfort inference algorithm. The algorithm runs daily and new predictive models are built every day. This happens because of the seasonality of the luminance target variable. Luminance values vary throughout the year, being low in the winter and quite high in the summer. It is impossible to have one model that can perform equally well all times of the year. The steps of the algorithm are briefly presented below (Figure 84):

- Data are retrieved from the pilots.
- Data are preprocessed.
- The models (day model, night model) are trained using Random Forest Regression.
- Luminance values are predicted for WVT pilot.
- Visual comfort is then calculated as described in Section 5.2.1.

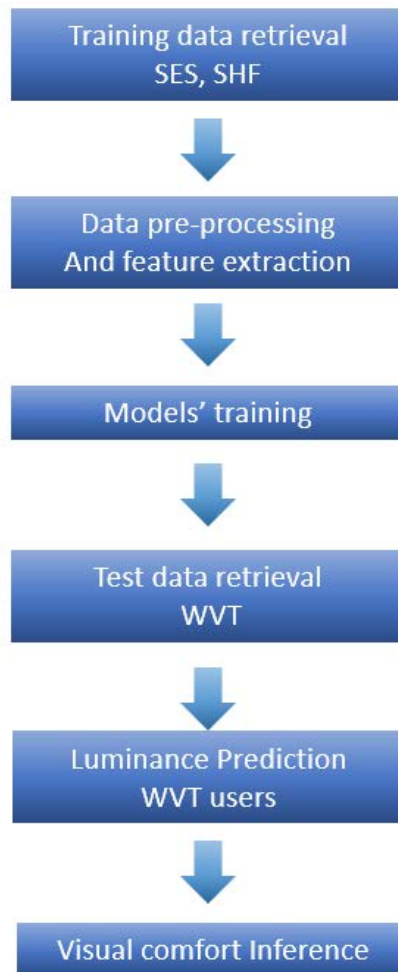


Figure 84: WVT visual comfort prediction flow chart.

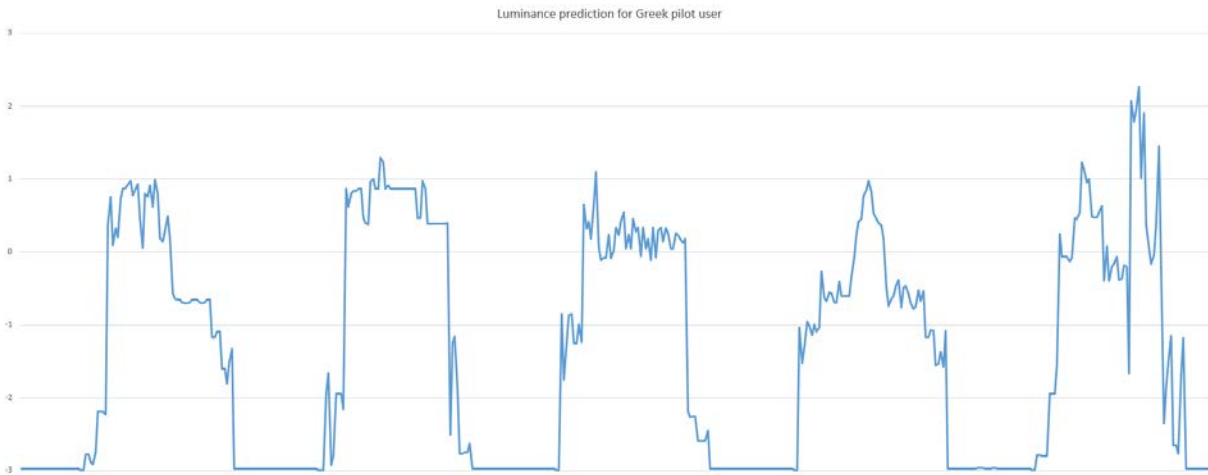


Figure 85: 5-day luminance prediction for WVT user

5.3 RESULTS

The calculation of thermal and visual comfort is performed in a daily basis. The results from the average thermal comfort and average visual comfort from September 2018 to March 2019, for the SES Pilot and SHF pilot, are depicted in Figure 86 and in Figure 87. Thermal comfort values are around 0 and Visual Comfort values are between -1 and -2. Visual comfort average is low because during night, at conditions of complete darkness (0 lux), visual comfort is constantly at -3.

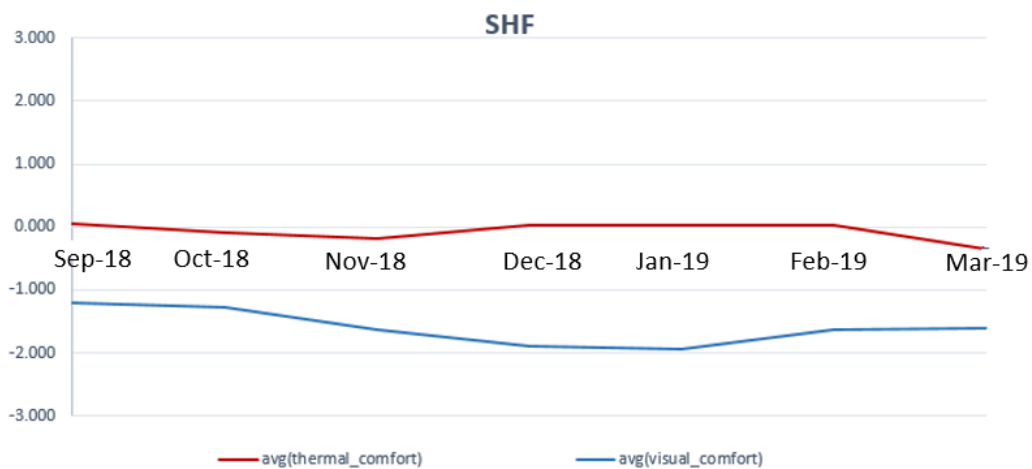


Figure 86: SHF average Thermal and Visual Comfort

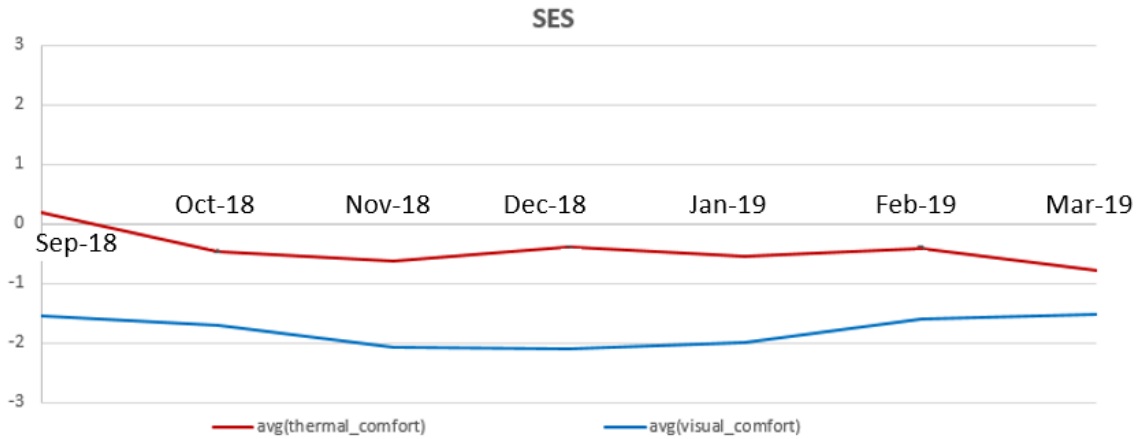


Figure 87: SES average Thermal and Visual Comfort

Although, the values of Thermal comfort are near zero there is an observable difference between the two pilots. This diversity can be effortlessly noticed in Figure 88.

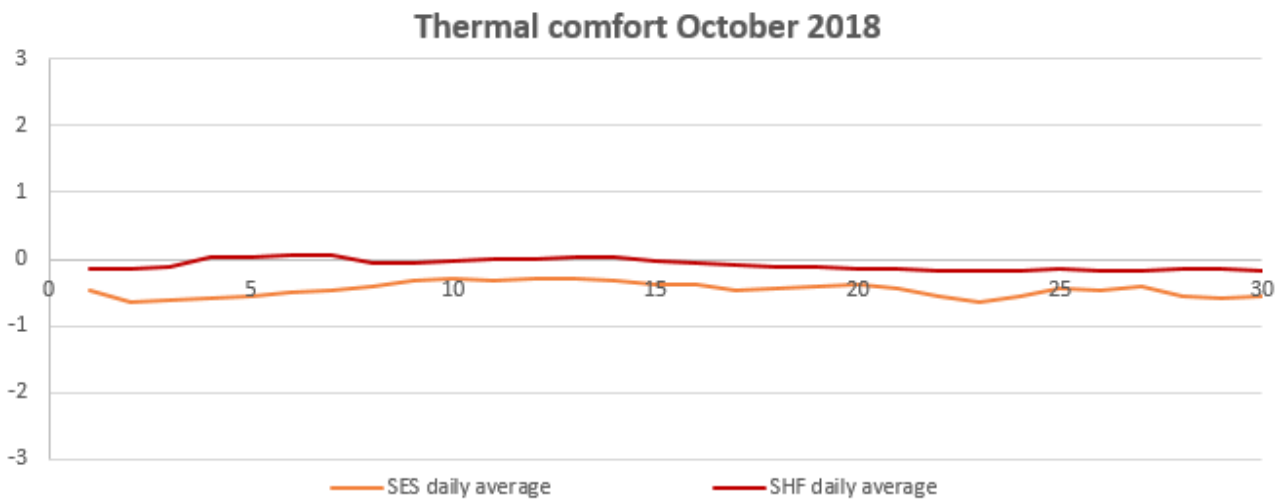


Figure 88: Daily Thermal Comfort for October 2018

A monthly example of thermal comfort from SES pilot is as depicted in Figure 89.

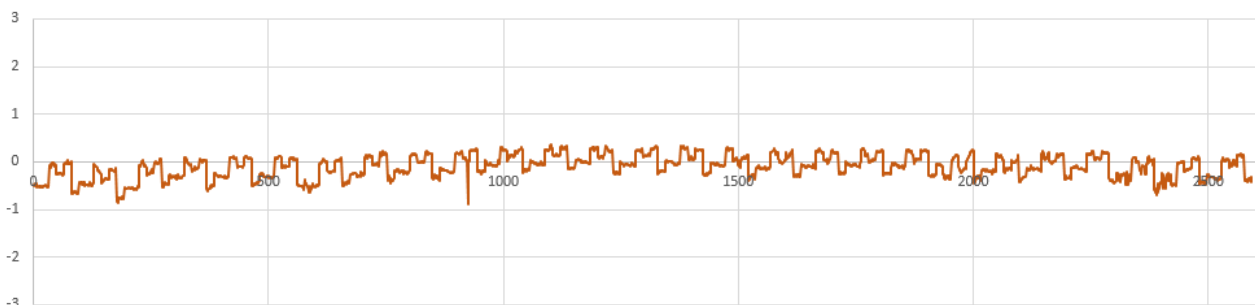


Figure 89: Monthly thermal comfort from SES pilot

A monthly example of visual comfort from SES pilot is as depicted in Figure 90.

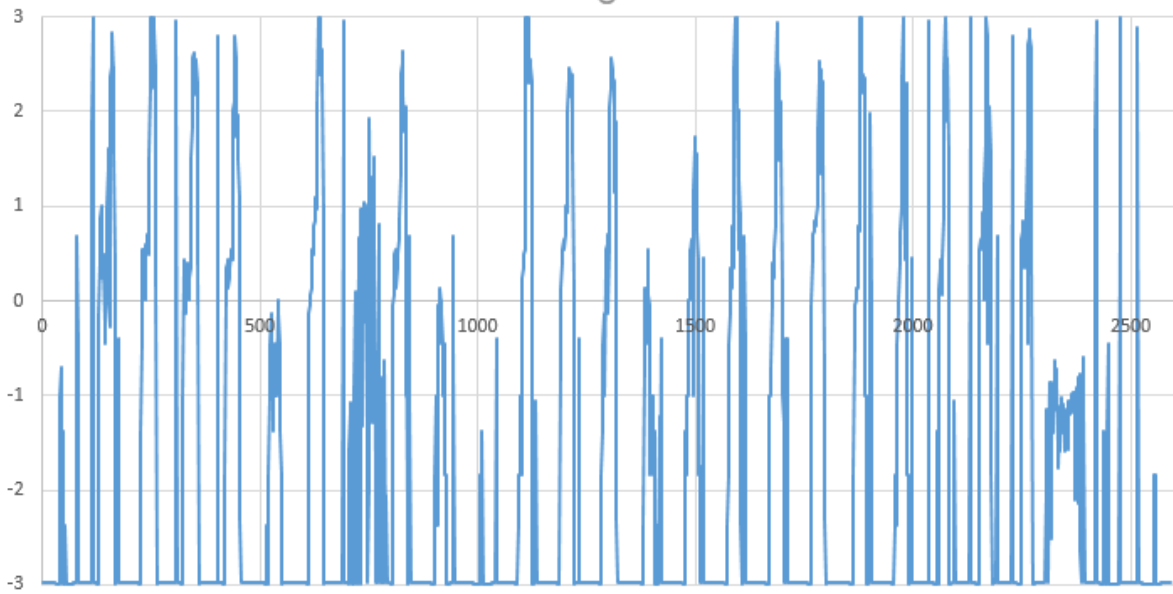


Figure 90: Monthly visual comfort from SES pilot

Weekly examples of thermal comfort and visual comfort from SES pilot are depicted in Figure 91 and in Figure 92.

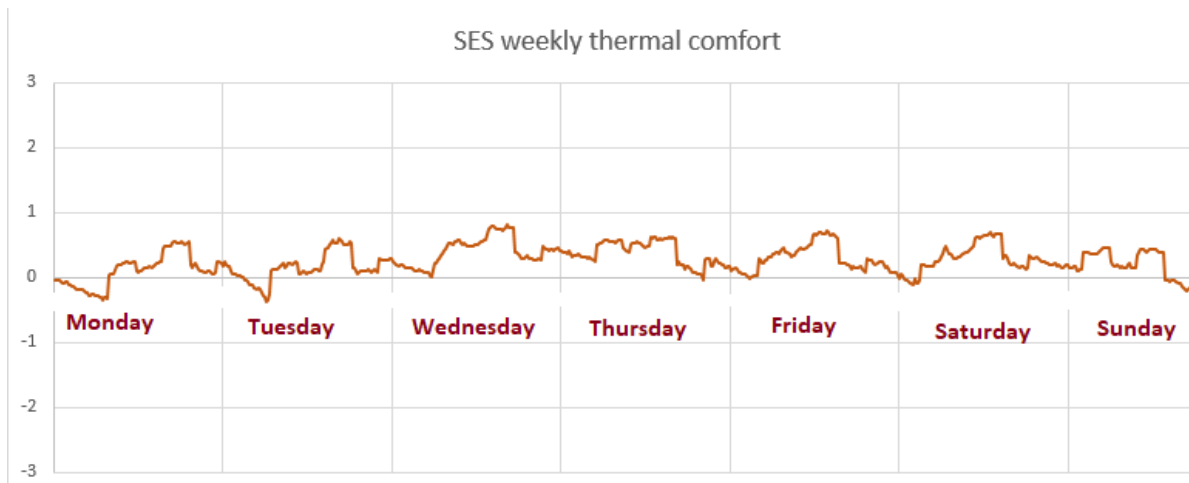


Figure 91: SES weekly thermal comfort

SES weekly visual comfort

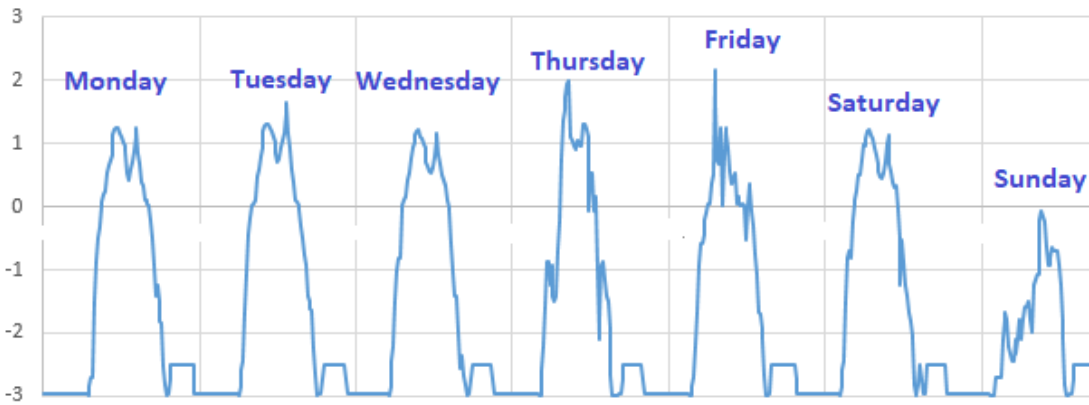


Figure 92: SES weekly visual comfort

Daily Example of thermal comfort and visual comfort of 5 different SES pilot users are depicted in Figure 93 and in Figure 94.

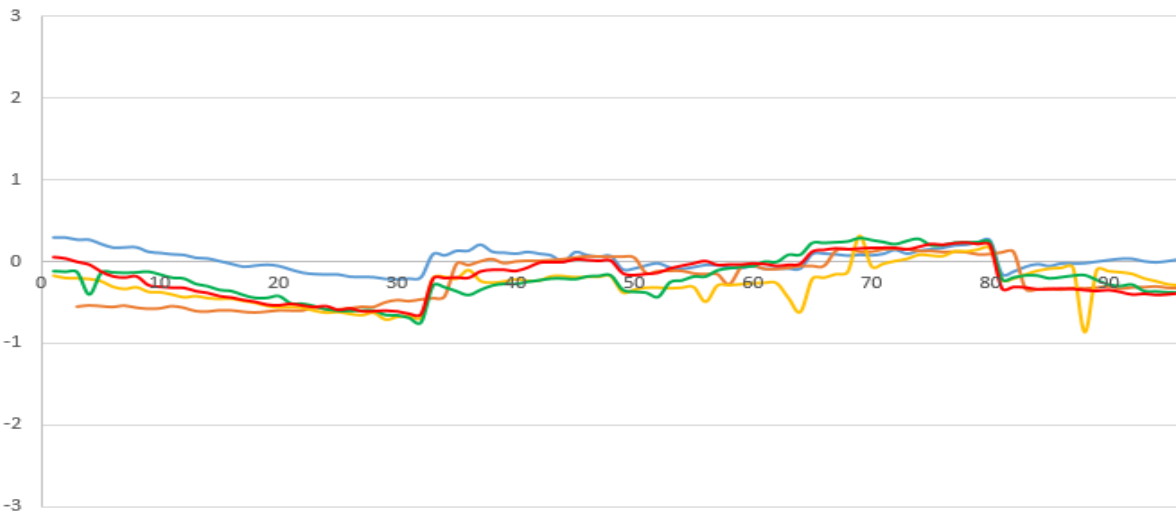


Figure 93: Daily thermal comfort of 5 different SES pilot users.

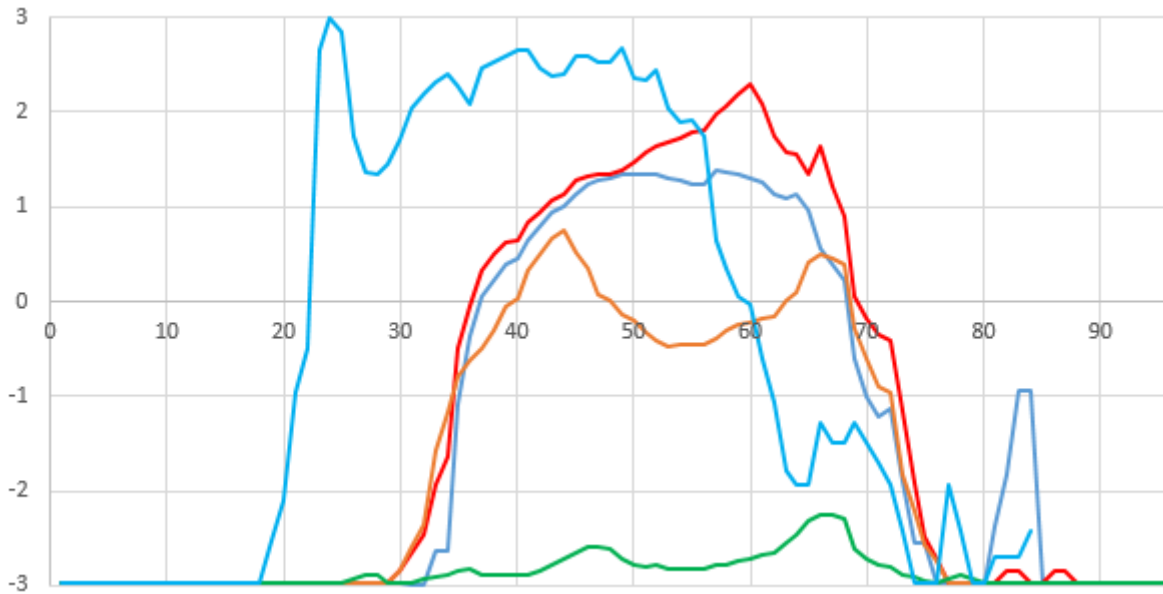


Figure 94: Daily visual comfort of 5 SES different pilot users.

Finally, an example of daily Thermal Comfort (96 timestamps) is presented in Figure 95 and daily visual comfort (96 timestamps) from SES pilot is presented in Figure 96.

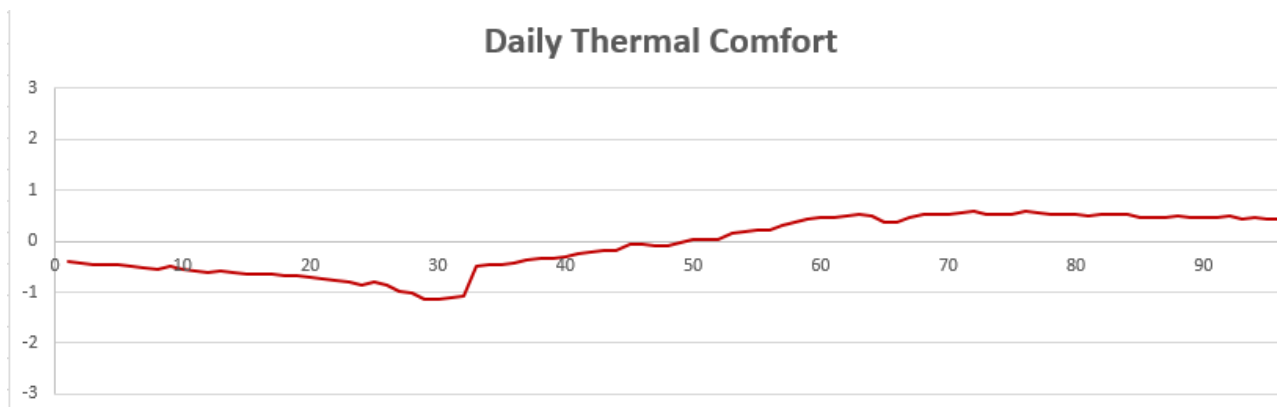


Figure 95: SES daily Thermal comfort

Daily Visual Comfort

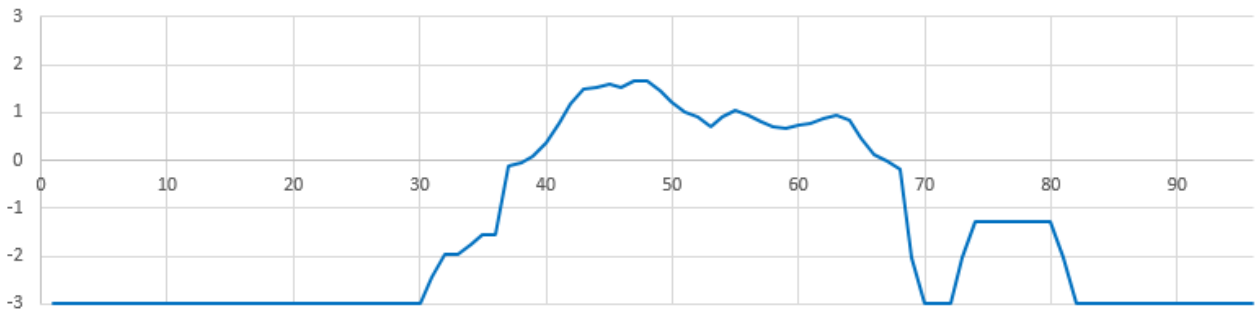


Figure 96: SES daily Visual comfort

6 CONCLUSIONS

This document describes the disaggregation approaches implemented in the final version of the algorithm adopted by the enCOMPASS platform to decompose the overall energy consumption pattern of a building in single end-uses. The proposed approach has been validated over synthetically generated and real energy consumption traces retrieved from a publicly available database and its performance has been evaluated for different granularities of the aggregated energy consumption measurements, showing that graceful degradation of the disaggregation results is achieved and that still accurate results can be obtained also in the case of data with 15-mins resolution (i.e. the data temporal resolution available in the enCOMPASS pilot case studies).

Furthermore, a Section of this document is dedicated to the description of the comfort inference engine developed within enCOMPASS project. Both thermal and visual comfort are estimated for all pilots. Advanced algorithms for training the algorithms to achieve a better convergence to the actual users' comfort have been utilized. In case of WV pilot, where luminance sensors were missing, the comfort values have been estimated based on training data created by the other two pilots. Some indicative and statistical results from the pilots are also presented in the document.

7 BIBLIOGRAPHY

- ANSI/ASHRAE Standard 55, Thermal Environmental Conditions for Human Occupancy, 1966.
- Baranski, M., & Voss, J. (2004). Genetic Algorithm for Pattern Detection in NIALM Systems. *Proceedings of IEEE International Conference on Systems Man and Cybernetics* (pp. 3462-3468). Hague, The Netherlands: IEEE.
- Batra, N., Kelly, J., Parson, O., Dutta, H., Knottenbelt, W., Rogers, A., . . . Srivastava, M. (2014). NILMTK: an open source toolkit for non-intrusive load monitoring. *Proceedings of the 5th international conference on Future energy systems* (pp. 265-276). Cambridge, United Kingdom: ACM.
- Camier, T. R., Giroux, S., Bouchard, B., & Bouzouane, A. (2013). Designing a NIALM in smart homes for cognitive assistance. *Procedia Computer Science*, 19, 524-532.
- Dong, R., Ratliff, L., Ohlsson, H., & Sastry, S. (2013). A dynamical systems approach to energy disaggregation. *IEEE 52nd Annual Conference on Decision and Control*.
- Figueiredo, M., Ribeiro, B., & de Almeida, A. (2013). On the regularization parameter selection for sparse code learning in electrical source separation. *Adaptive and Natural Computing Algorithms*.
- Hart, G. (1992). Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12).
- Jama : A java matrix package*. (2017, 09). Retrieved from <http://math.nist.gov/javanumerics/jama/>
- Johnson, M., & Willsky, A. (2013). Bayesian nonparametric hidden semi Markov models. *The Journal of Machine Learning Research*, 14(1).
- Likas, A., Vlassis, N., & Verbeek, J. (2003). The global k-means clustering algorithm. *Pattern recognition*, 36(2), 451-461.
- Makonin, S., Popowich, F., Bartram, L., Gill, B., & Bajic, I. (2013). AMPds: a public dataset for load disaggregation and eco-feedback research. *IEEE Electrical Power and Energy Conference (EPEC)*, (pp. 1-6).
- Marchiori, A., Hakkarinen, D., Han, Q., & Earle, L. (2011). Circuit-level load monitoring for household energy management. *IEEE Pervasive Computing*, 10(1), 40-48.
- Mysql: The world's most popular open source database*. (2017, 09). Retrieved from <http://dev.mysql.com/>
- Parson, O., Ghosh, S., Weal, M., & Rogers, A. (2012). Non-Intrusive Load Monitoring using prior models of general appliance types. *AAAI Conference on Artificial Intelligence*.
- Piga, D., Cominola, A., Giuliani, M., Castelletti, A., & Rizzoli, A. E. (2016, May). Sparse Optimization for Automated Energy End Use Disaggregation. *IEEE Transactions on Control Systems Technology*, 24(3), 1044-1051.
- Srinivasan, D., Ng, W., & Liew, A. (2006). Neural-network-based signature recognition for harmonic source identification. *IEEE Transactions on Power Delivery*, 21(1).
- Suzuki, K., Inagaki, S., Suzuki, T., Nakamura, H., & Ito, K. (2008). Nonintrusive appliance load monitoring based on integer programming. *Proceedings of SICE Annual Conference*.

Zeifman, M., & Roth, K. (2011). Nonintrusive appliance load monitoring: Review and outlook. *IEEE Transactions on Consumer Electronics*, 57(1).

Zia, T., Bruckner, D., & Zaidi, A. (2011). Hidden Markov Model based procedure for identifying household electric loads. *37th Annual Conference on IEEE Industrial Electronics Society*,.

Zoha, A., Gluhak, A., Imran, M., & Rajasegarar, S. (2012). Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey", *Sensors*, 12(12).