ECSEL Joint Undertaking

Electronic Components and Systems for European Leadership

MANTIS

Cyber Physical System based Proactive Collaborative Maintenance

D1.2 Consolidated State-of-the-Art of Sensor-based Proactive Maintenance

Appendix 15:

Modelling techniques for high-frequency time-series data

Work Package	WP1 - Service platform architecture requirement definition. Scenarios and use cases descriptions
Version	1.0
Contractual Date of Delivery	30/04/2016
Actual Date of Delivery	03/06/2016
Dissemination Level	Public
Responsible	Erkki Jantunen
Contributors	Tom Tourw (SIR), Elena Tsiporkova (SIR), Mathias Verbeke (SIR), Tom Ruette (SIR), Csaba Heged űs (A

The MANTIS consortium consists of:

Num.	Short Name	Legal Name	Role	Country
1	MGEP	Mondragon GoiEskolaPoliteknikoa J.M.A. S.Coop.	CO	ES
2	MONDRAGON	Mondragon CorporacionCooperativaS.Coop.	BEN	ES
3	IKERLAN	IkerlanS.Coop.	BEN	ES
4	TEKNIKER	FundacionTekniker	BEN	ES
5	FARR	FagorArrasateS.Coop.	BEN	ES
5.1	KONIKER	KonikerS.Coop.	TP	ES
6	GOIZPER	GoizperS.Coop.	BEN	ES
7	ACCIONA	Acciona Infraestructuras S.A.	BEN	ES
8	MSI	Mondragon Sistemas De InformacionS.Coop.	BEN	ES
9	VTT	TeknologianTutkimuskeskusVTT Oy	BEN	FI
10	LUAS	Lapin AmmattikorkeakouluOy	BEN	FI
11	NOME	Nome Oy	BEN	FI
12	FORTUM	Fortum Power And Heat Oy	BEN	FI
13	SQ	SolteqOyj	BEN	FI
14	WAPICE	WapiceOy	BEN	FI
15	AAU	Aalborg Universitet	BEN	DK
16	DANFOSS	Danfoss A/S	BEN	DK
17	UNIV	Universal Foundation A/S	BEN	DK
18	HGE	Hg Electric A/S	BEN	DK
19	VESTAS	Vestas Wind Systems A/S	BEN	DK
20	SIRRIS	Sirris Het Collectief Centrum Van De TechnologischeIndustrie	BEN	BE
21	ILIAS	Ilias Solutions Nv	BEN	BE
22	ATLAS	Atlas Copco Airpower Nv	BEN	BE
23	3E	3e Nv	BEN	BE
24	PCL	Philips Consumer Lifestyle B.V.	BEN	NL
25	PHC	Philips Medical Systems Nederland B.V.	BEN	NL
26	PHILIPS	Philips Electronics Nederland B.V.	BEN	NL
27	S&T	Science and Technology B.V.	BEN	NL
28	TU/E	TechnischeUniversiteit Eindhoven	BEN	NL
29	RUG	Rijksuniversiteit Groningen	BEN	NL
30	UNINOVA	UNINOVA - Instituto de Desenvolvimento de Novas Tecnologias	BEN	PT
31	ISEP	Instituto Superior de Engenhariado Porto	BEN	PT
32	INESC	Instituto de Engenharia de Sistemas eComputadores do Porto	BEN	PT
33	ADIRA	ADIRA - Metal Forming Solutions S.A.	BEN	PT
34	ASTS	Ansaldo STS S.p.A.	BEN	IT
35	CINI	Consorzio Interuniversitario Nazionale per l'Informatica	BEN	IT
36	AIT	Austrial Institute of Technology GmbH	BEN	AT
37	HBM	HottingerBaldwniMesstechnik GmbH	BEN	AT
38	INNOTEC	Innovative Technology and Science Limited	BEN	UK
39	AITIA	AITIA International Inc.	BEN	HU
40	BME	Budaperst University of Technology and Economics	BEN	HU
41	JSI	Josef Stefan Institute	BEN	SI
42	XLAB	XLAB d.o.o.	BEN	SI
43	FHG	Fraunhofer Institute for Experimental Software Engineering IESE	BEN	DE
44	M2X	M2Xpert GmbH & Co KG	BEN	DE
45	STILL	STILL GMBH	BEN	DE
46	BOSCH	Robert Bosch GMbH	BEN	DE
47	LIEBHERR	Liebherr-Hydraulikbagger GmbH	BEN	DE

Document Revisions & Quality Assurance

Revisions:

Version	Date	By	Overview
0.1	6.8.2015	Mathias Verbeke (SIR)	First draft
0.2A	18.8.2015	Csaba Heged űs (	Chapter on Statistical Analysis added
0.3	10. 09. 2015	Csaba Heged űs (	Fitted the contributions into the document and reference list
0.4	06.10.2015	Riku Salokangas	Added contractual date of delivery etc.
0.5	30.03.2015	Tom Ruette (SIR)	Related standards
1.0	02/06/2016	Mikel Muxika (MGEP)	Format correction Deliverable info update

Abstract

In this appendix, a number of modelling techniques for high-frequency time-series data will be described.

The first set of methods is situated in the domain of data stream mining, focusing finding (frequent) patterns of events in the data stream and predicting upcoming events.

Second, cluster analysis techniques estimate the similarity between objects and propose a categorization into k clusters.

A third set of methods is related to time-varying system identification, which tries to identify the system behind a series of states in which the system is varying over time due to an unobserved scheduling parameter.

In Section 4, probabilistic modeling techniques are discussed, which describe the data that can be observed from a system, using probability theory to express the uncertainty resulting from noise and missing data; issues that are often inherent when working with sensor data. Subsequently, probabilistic reasoning can be used to infer additional knowledge and make predictions.

Finally, we present a reminder about statistical analysis on time series data that might be useful for detecting various types of very long term tendencies. These methods are mainly used in econometrics and financial analysis, but could be relevant to issues that arise in MANTIS.

Table of Contents

1 Pattern discovery and pattern-based prediction

Data Stream Mining is the process of extracting useful information from continuously incoming data. A data stream is an ordered sequence of events that can typically be read only once by an algorithm, and that is too large to be stored for later analysis. The main challenges for finding interesting patterns in data streams are:

1. to instantly output up-to-date patterns
2. to keep the memory consumption within reasonable limits by summarizing the part of the stream that has already been seen
3. to keep the results as correct as possible when deciding that parts of the data seen in the past can be forgotten (e.g. concept drift)

The earliest methods for finding patterns in streams focused on finding frequent itemsets. An itemset is a pattern consisting of a set of events, where the order in which the events occur is not important [1]. Typically, a data stream is assumed to consist of incoming transactions, each of which contains a number of items (or events). The goal is to find sets of items that often occur together in the transactions making up the stream. The existing methods can be classified in many ways. First of all, the adopted approaches can be divided into the false-positive methods and the false-negative methods.

The false positive approach guarantees that all frequent itemsets will be found, but at a cost of discovering some itemsets that are in fact infrequent. One of the best-known false-positive methods is lossy counting [2]. Following on this work, further attempts have been made to improve on both performance and accuracy of the proposed algorithm [3, 4, 5, 6].

The false negative approach gives a guarantee that no infrequent itemset will find its way into the output, but some frequent itemsets may be missed as well [7]. Recently, a probabilistic lossy counting method has been proposed that tries to balance between the two approaches [8].

A second classification of the existing methods can be made based on how the incoming stream is processed. Update-per-transaction methods [3, 5, 6] instantly process events as soon as they occur, while update-per-batch methods [2, 4] process events in bulk. Clearly, while the former approach is capable of producing better results at any given moment, the latter approach results in higher efficiency.

Naturally, itemsets are not the only patterns that can be discovered in streams. A sequential pattern [9] consists of a set of events which reoccur in a given order. Here, therefore, the order in which the events occur in the stream does matter. There exists a large body of work in this field [10, 11, 12, 13, 14, 15, 16, 17, 18, 19], which can again be classified in a similar way as above.

However, neither itemsets nor sequential patterns can express all the possible interactions between events that can occur in a stream. A much richer type of pattern are episodes [20, 21, 22], which can express partial orders between the events making up a pattern. For example, a pattern "event A is followed by events B and C, which are then followed by event D", where the order of B and C can vary, can be expressed as an episode, but not as either a sequential pattern or an itemset. Discovering such partial orders in a data stream has not yet been done.

Most existing pattern mining work in data stream mining has gone into finding frequent patterns. However, frequency is certainly not the only possible quality measure that can be used to evaluate a pattern. Recently, the "cohesion" quality measure has been proposed in order to evaluate itemsets consisting of events occurring in a sequence of data [23]. Here, cohesion is defined as the average minimal window (or time span) containing the entire itemset. The advantage of cohesion is that it rewards more cohesive patterns even if they are less frequent than some other less cohesive patterns, and that it also does not require a subjective user-defined parameter such as a sliding window length [20], like many other algorithms.

Finally, some efforts have been made to predict upcoming events in a stream, but existing work is rather limited. For example, Hidden Markov Models based on frequent episodes mined from historical data have been used for a prediction task in a stream [24]. However, this method can only predict whether a predefined target event type will occur, and is not suitable for predicting arbitrary events. Two algorithms, DeMO and CBS-Tree [25], attempt to make predictions using episode rules over event streams. A rule is matched based on the latest minimal occurrence of the antecedent of the rule, and the consequent event is then predicted. A further approach attempts to match multiple episode rules for stream prediction by proposing an algorithm to generate all representative episode rules based on frequent closed episodes [26]. However, both of these approaches use episode rules discovered in historical data, while, in a streaming environment, we need to get the latest information from the current data.

2 Cluster analysis for time series data

Cluster analysis techniques estimate the similarity between objects and propose a categorization into *k* clusters by comparing the distance of the object to be clustered to the existing clusters; the object that needs to be clustered is assigned to the nearest cluster [27]. An important aspect of cluster analysis is the distance metric that is used to obtain the similarity between objects. Several distance metrics have been proposed, e.g. Euclidean and Minkowski [27], but these typically do not take into account that the object characteristics can be time-ordered. When objects are characterized by time-ordered data points, distance metrics that assume that the *i*-th point of one time series with the *i*-th point of the other may produce a poor similarity score. It is better to use a non-linear or elastic alignment to have a more intuitive similarity measure, allowing similar shapes to match even if they are out of phase in the time axis (cf. Figure 1). This alignment procedure is known as Dynamic Time Warping [28], which is also used in data synchronisation.

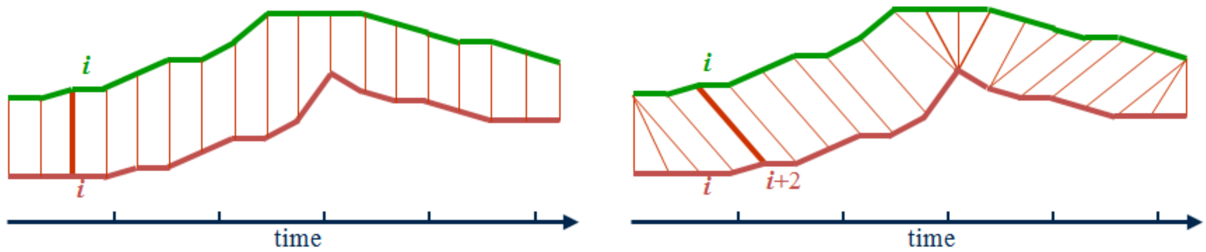

Figure 1 Dynamic Time Warping for distance measurement between two time series data streams

We propose to use a distance metric that incorporates Dynamic Time Warping as the distance metric that underlies a cluster analysis. Cluster analysis can be used to detect groups of near-identical systems in a network or fleet of collaborative systems that behave similar, but differently from the other groups. The existence of clear system clusters may be a consequence of certain external influence, e.g. orientation of solar panels with respect to the sun. However, if external influences can be discarded, it may also be indicative of unexpected deviating performance behavior between presumably identical systems.

3 Time-varying system identification

System identification for linear time-varying systems is a generalization of the identification for parameter-varying system identification. In parameter-varying system identification it is assumed that the system properties are defined by a scheduling parameter, e.g., for a crane hoisting a mass, the resonance frequency of the pendulum motion of the mass is defined by the cable length. During hoisting, the cable length varies and thus affects the resonance frequency. Time-varying system identification starts from the same point of view but considers that the system is varying over time due to an unobserved scheduling parameter, e.g., when the cable length during hoisting is unknown. In practice, modeling a system as time-varying is a more general approach to the parameter-varying system identification, which requires an estimate of the scheduling parameter. When the scheduling parameter is unknown, or not (adequately) measured, a time-varying system identification remains possible.

Several approaches exist to model parameter- or time-varying systems. A pragmatic method is to use sliding windows to process consecutive short time records of the investigated system [29, 30]. Each time record is processed as it was a time-invariant system. By sliding the window to consecutive time instances and forgetting older results, an approximation of the time-varying system, built up from short time-invariant models, is obtained. However, as such a method still processes each time record as a time-invariant system the continuous time-varying nature is ignored causing an intrinsic delay in the identification [31].

One possible alternative is to use parametric approaches that use either time-domain state-space models with time-varying state space matrices, a.k.a. the polytopic state-space model [32], or within the frequency domain where the model coefficients are considered as time-varying [33]. In these approaches the state space matrices or the frequency domain coefficients are modelled as parametric models in time, e.g., Legendre-polynomials and the time-varying nature of the system is part of the model.

4 Probabilistic modelling techniques

Next to events, raw sensor data is typically noisy, continuous multi-stream data and also on this level a wide range of mathematical and statistical algorithms are available to build models. Probabilistic modelling techniques describe the data that can be observed from a system, using probability theory to express the uncertainty resulting from noise and missing data. Subsequently, probabilistic reasoning can be used to infer additional knowledge and make predictions.

Some well-known algorithms to build predictive models are neural networks, logistic regression, decision trees, Bayesian networks, Kriging, Support Vector Machines, rational functions, splines, radial basis functions, various classifiers and gradient-based online learning algorithms [33]. Time series models like Box-Jenkins, auto-regressive integrated moving average models and adaptive neuro-fuzzy inferencing systems, are also used to predict future behavior of a system by taking the autocorrelation, trends or periodic variations of temporal data streams into account [34]. Multiple model types can be run in concert and/or model ensembles can be built. Several approaches for the hyperparametrization of models are available as well, e.g., Efficient Global Optimization, meta-heuristics, evolutionary algorithms such as Particle Swarm Optimization, genetic algorithms [35] or simulated annealing. Model selection can be done using algorithms like cross validation, Akaike□s Information Criterion, Bayesian Information Criterion or a Leave-out set. The choice of appropriate algorithms is strongly dependent on the problem at hand [36], and needs careful consideration within this project. It is also investigated how these algorithms can be made robust with respect to large stochastic variability that is inherent to sensor data, or the presence of latent variables that cannot be monitored by the sensors that are present in the fleet. The above mentioned modeling algorithms also assume that the data stream is continuous and uninterrupted. The use of probabilistic modelling techniques that can capture the relationships between streams and derived features while being robust to missing data are also being examined. This will keep the model effective when sensors fail.

In particular, Probabilistic Graphical Models (PGM) have grown in popularity due to their interpretability and elegant visualization. The main constraint was, and is, the complexity of inference. Where originally, only inference in tree-like representations was feasible, this has been upgraded to low-threewidth graphical models to allow more complex interactions in a model as a result of improved inference algorithms [37]. For example, this was the premise of Fault Tree Analysis for dependability analysis and has been generalized to Bayesian networks [38]. In general, however, it is intractable to perform exact inference in PGM. For this reason one typically resorts to sampling. The disadvantage of approximate inference is the difficulty to detect whether the result is close to converging or not. An alternative approach is to learn only graphs that guarantee tractable, thus efficient, inference. For example, by limiting the allowed treewidth of the graph [39, 40]. A recent trend is to loosen the treewidth constraint to models that can be represented by an arithmetic circuit (e.g. d-DNNF, SDD, SPN) [41, 42]. These models allow a higher level of complexity while guaranteeing tractable inference.

5 Statistical Analysis of Time Series Data

There might be use for MANTIS in implementing long term statistical analysis using similar techniques as in for example economic statistics (econometrics). These methods are used for example in high frequency trading (in stock markets), creating global statistical databases (e.g. EuroStat) or market research for product entry.

The following summary will give a reminder about the basic techniques of statistical time series analysis, give references for catching-up and discovering how it might be rewarding to research and implement these approaches in MANTIS as well. There are several software tools that have already implemented and verified.

These methodologies require pre-processed (basically query-able) time series data sets. That means the data is registered with its timestamp(not necessarily equidistantly captured) and is available in high resolution within a very wide timeframe (long term data). Let's assume that such databases are present containing information sent in by devices of a MANTIS implementation.

Different types of very slow underlying issues could be detected, for example item wear, asset efficiency degradation, sensory precision loss (calibration drifts) or changes in the production environment (even human efficiency changes), while forecasting from trend analysis is also commonly used.

There is wide literature and available methodologies on analysing (high frequency) time series data. There are several approaches to analysing time series data in economic statistics:

- Basic analysis using chronological averages (on stock and flow typed time series), spread-like or element-difference based metrics (measuring volatility)
- ARMA-ARIMA based (Auto Regressive Integrated Moving Average) models (stochastic approaches, like the Box-Jenkins-model [44], mostly used for short term analysis), for a detailed introduction see [43]
- Frequency- decomposition models (spectral analysis, wavelet analysis [49])
- Time series regression models (regression models fitted on time series data)

Basic analysis can be useful, for example when the measured data is mostly constant or has thresholds, like measuring different asset stocks or non-volatile trends.

There are several ARMA based models. For a detailed introduction to the field, see [47] or [48].

The decomposition models (as their name show) decompose time series (in basic econometric models) using the following cyclical natural factors of productivity:

- very long term slow trends (e.g. actual production issues caused by underlying problems)
- quarterly seasonality: the natural fluctuation of supply-demand and production (e.g. the summer is always less productive than the winter and there is nothing wrong with that)
- irregular cyclical component (for example temporary issues)
- accidental error (measurement error or by chance)

These models are mostly based on the Fourier transform. For a detailed summary of the field, see [45].

As the time series are often non-repeatable measurements, using these models often face problems caused by measurement errors and noise. These issues can be mitigated by using statistical filtering and smoothing methods, see [8] for summary of post K□lm□n-filtering techniques.

The most basic technique is trend analysis: linear, polynomial, exponential, logistical or their combinations, even ones reaching a saturation point. These trends can also be used for forecasting (ex ante).

These use basic MSE (Mean Square Error) approach, but there are other, more sophisticated filters, like Hodrick-Prescott filtering. It is used to obtain a smoothed-curve representation of a time series, one that is more sensitive to long-term than to short-term fluctuations. For further work on statistical filtering, see [50].

Statistical hypothesis testing might also come in handy in the analysis of long term time series data. These approaches might be useful for checking different parameters of partial time series as a population, giving estimations on a whole process as population (e.g. production lines), or looking for changes in the parameters of a population based on samples. The key words here are significance level, population, (independent) sample(s), null and alternative hypothesis, test functions, distribution functions, see [46].

- Tests about statistical parameters based on samples (e.g. determining practical useful life expectancies based on previous samples)
- Tests for determining if sample follows a specific distribution (e.g. Pearson□s Chi Square test) or if two samples are from the same distribution (e.g. Kolmogorov-Smirnov test)
- Tests for determining population ratios (e.g. faulty products ratio based on multiple independent samples)
- Testing probabilistic independence between multiple samples

Here are some use cases where these approaches would be invaluable:

- sensory input outage, missing data (e.g. with ARMA models)
- Outlier data
- Sensory drift, loss of calibration (e.g. with long term trend analysis)
- Process deviation from samples (e.g. with distribution testing hypothesis)
- Life expectancy from item sets
- Faulty item ratios (e.g. with population ratio testing with samples)

6 Related standards

- iec 61649: Weibull analysis

- IEC 61703: Mathematical expressions for reliability, availability, maintainability and maintenance support terms

- IEC 61165: Application of Markov techniques

References

[1] Rakesh Agrawal, Tomasz Imielinski, Arun N. Swami: Mining Association Rules between Sets of Items in Large Databases. SIGMOD Conference 1993: 207-216

[2] Gurmeet Singh Manku, Rajeev Motwani: Approximate Frequency Counts over Data Streams. VLDB 2002: 346-357

[3] JoongHyuk Chang, Won Suk Lee: Finding recent frequent itemsets adaptively over online data streams. KDD 2003: 487-492

[4] Chris Giannella, Jiawei Han, Jian Pei, Xifeng Yan, Philip S. Yu: Mining frequent patterns in data streams at multiple time granularities. Next generation data mining 212 (2003): 191-212

[5] Yun Chi, Haixun Wang, Philip S. Yu, Richard R. Muntz: Catch the moment: maintaining closed frequent itemsets over a data stream sliding window. Knowl. Inf. Syst. 10(3): 265-294 (2006)

[6] ToonCalders, NeleDexters, Bart Goethals: Mining Frequent Itemsets in a Stream. ICDM 2007: 83-92

[7] Jeffrey Xu Yu, Zhihong Chong, Hongjun Lu, Aoying Zhou: False Positive or False Negative: Mining Frequent Itemsets from High Speed Transactional Data Streams. VLDB 2004: 204-215

[8] Xenofontas A. Dimitropoulos, Paul Hurley, Andreas Kind: Probabilistic lossy counting: an efficient algorithm for finding heavy hitters. Computer Communication Review 38(1): 5 (2008)

[9] Rakesh Agrawal, RamakrishnanSrikant: Mining Sequential Patterns. ICDE 1995: 3-14

[10] JoongHyuk Chang, Won Suk Lee: Efficient mining method for retrieving sequential patterns over online data streams. J. Information Science 31(5): 420-432 (2005)

[11] Gong Chen, Xindong Wu, Xingquan Zhu: Sequential Pattern Mining in Multiple Streams. ICDM 2005: 585-588

[12] ChedyRa□ssi, Pascal Poncelet, MaguelonneTeisseire: Need For Speed: Mining Sequential Patterns in Data Streams. BDA 2005

[13] Alice Marascu, FlorentMasseglia: Mining sequential patterns from data streams: a centroid approach. J. Intell. Inf. Syst. 27(3): 291-307 (2006)

[14] Chin-Chuan Ho, Hua-Fu Li, Fang-FeiKuo, Suh-Yin Lee: Incremental Mining of Sequential Patterns over a Stream Sliding Window. ICDM Workshops 2006: 677-681

[15] C. I. Ezeife, MostafaMonwar: SSM: A Frequent Sequential Data Stream Patterns Miner. CIDM 2007: 120-126

[16] Haifeng Li, Hong Chen: GraSeq: A Novel Approximate Mining Approach of Sequential Patterns over Data Stream. ADMA 2007: 401-411

[17] Lei Chang, Tengjiao Wang, Dongqing Yang, Hua Luan: SeqStream: Mining Closed Sequential Patterns over Stream Sliding Windows. ICDM 2008: 83-92

[18] Luiz F. Mendes, Bolin Ding, Jiawei Han: Stream Sequential Pattern Mining with Precise Error Bounds. ICDM 2008: 941-946

[19] Syed KhairuzzamanTanbeer, Chowdhury Farhan Ahmed, Byeong-SooJeong, Young-Koo Lee: Efficient frequent pattern mining over data streams. CIKM 2008: 1447-1448

[20] HeikkiMannila, HannuToivonen, A. InkeriVerkamo: Discovery of Frequent Episodes in Event Sequences. Data Min. Knowl. Discov. 1(3): 259-289 (1997)

[21] NikolajTatti, Boris Cule: Mining closed strict episodes. Data Min. Knowl. Discov. 25(1): 34-66 (2012)

[22] Boris Cule, NikolajTatti, Bart Goethals: MARBLES: Mining association rules buried in long event sequences. Statistical Analysis and Data Mining 7(2): 93-110 (2014)

[23] Boris Cule, Bart Goethals, C□line Robardet: A New Constraint for Mining Sets in Sequences. SDM 2009: 317-328

[24] SrivatsanLaxman, VikramTankasali, Ryen W. White: Stream prediction using a generative model based on frequent episodes in event sequences. KDD 2008: 453-461

[25] Chung-Wen Cho, Yi-Hung Wu, Show-Jane Yen, Ying Zheng, Arbee L. P. Chen: On-line rule matching for event prediction. VLDB J. 20(3): 303-334 (2011)

[26] Huasheng Zhu, Peng Wang, Wei Wang, Baile Shi: Stream Prediction Using Representative Episode Rules. ICDM Workshops 2011: 307-314

[27] Ricardo A. Baeza-Yates, Berthier A. Ribeiro-Neto: Modern Information Retrieval. ACM Press / Addison-Wesley 1999

[28] David Sankoff, Joseph B. Kruskal: Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison. Addison Wesley (1983)

[29] Lennart Ljung, TorstenS□derstr□m: Theory and Practice of Recursive Identification. MIT Press 1983

[30] MaciejNied źwiecki: Identification of Time-varying Processes. J. Wiley & Sons 2000

[31] MaciejNiedzwiecki: Optimal and suboptimal smoothing algorithms for identification of time-varying systems with randomly drifting parameters. Automatica 44(7): 1718-1727 (2008)

[32] Atsushi Fujimori, Lennart Ljung: Parameter estimation of polytopic models for a linear parameter varying aircraft system. Transactions of the Japan Society for Aeronautical and Space Sciences 49(165): 129-136 (2006)

[33] Christopher M. Bishop: Pattern Recognition and Machine Learning (information science and statistics). Springer (2006)

[34] Allan D. R. McQuarrie, Chih-Ling Tsai: Regression and Time Series Model Selection. World Scientific (1998)

[35] Yu Xinjie: Introduction to Evolutionary Algorithms. Computers and Industrial Engineering 2010: 1

[36] Dirk Gorissen, Ivo Couckuyt, Piet Demeester, Tom Dhaene, KarelCrombecq: A Surrogate Modeling and Adaptive Sampling Toolbox for Computer Based Design. Journal of Machine Learning Research 11: 2051-2055 (2010)

[37] Andrea Bobbio, Luigi Portinale, Michele Minichino, Ester Ciancamerla: Comparing Fault Trees and Bayesian Networks for Dependability Analysis. SAFECOMP 1999: 310-322

[38] Adnan Darwiche: Modeling and Reasoning with Bayesian Networks. Cambridge University Press (2009)

[39] Anton Chechetka, Carlos Guestrin: Efficient Principled Learning of Thin Junction Trees. NIPS 2007: 273-280

[40] Zhu Yongli, HuoLimin, Lu Jinling: Bayesian networks-based approach for power systems fault diagnosis. IEEE Transactions onPower Delivery 21(2): 634-639 (2006)

[41] Daniel Lowd, AmirmohammadRooshenas: Learning Markov Networks With Arithmetic Circuits. AISTATS 2013: 406-414

[42] Robert Gens, Pedro M. Domingos: Learning the Structure of Sum-Product Networks. ICML (3) 2013: 873-880

[43] Brockwell, P. J.; Davis, Richard A. (2002): *Introduction to Time Series and Forecasting*. Second Edition, Springer Verlag, ISBN 0-387-95351-5. (Available: http://www.masys.url.tw/Download/2002-Brockwell-Introduction%20Time%20Series%20and%20Forecasting.pdf Accessed: 2015. 08. 25.)

[44] Box, George; Jenkins, Gwilym (1970): *Time Series Analysis: Forecasting and Contro*l. San Francisco, Holden-Day.

[45] Steehouwer, H. (2009): *A Frequency Domain Methodology for Time Series Modeling.* OFRC Methodological paper No. 2008-02, May 2009. Available: http://nl.ortec-finance.com/~/media/Files/Research/Working%20Papers/Methodological/Public/OFRC%20Methodological%202008-02.ashx Accessed: 205. 08. 25.)

[46] Walpole R.E. et al (2011): *Essentials of Probability & Statistics for Engineers & Scientists*. Pearson, USA, ISBN 0-321-78373-5.

[47] Cromwell, Jeff B. et al. (1994): *Multivariate Tests for Time Series Models*. SAGE Publications. ISBN 0-8039-5440-9.

[48] Kirchg□sser, G.; Wolters, J.; Hassler, U. (2013): *Introduction to Modern Time Series Analysis*. Second Edition. Springer Verlag, Berlin. ISBN 978-3-642-33435-1.

[49] Abramovich, F.; Bailey, T.C.; Sapatinas, T.S. (2000): *Wavelet analysis and its statistical applications*. The Statistician Vol 49. , Part 1, pp. 1-29.

[50] Moore, J.B. (1977): *Statistical Filtering. In: Mathematical Control Theory*, pp. 152-169. Springer Verlag, Berlin. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.384.6363&rep=rep1&type=pdf Accessed: 2015. 08. 26.)