SYSTEMATIC COMPARISON OF EXISTING AND NEW APPROACHES FOR MONITORING COMPLIANCE RULES OVER BUSINESS PROCESSES

Assoc.Prof. eng. Nikolova-Alexieva V. PhD.[1], Assoc. Prof. Mihova T. PhD.[2],
Faculty of Economics – University of Food Technology-Plovdiv, Bulgaria [1]
Faculty of Mechanical Engineering - Technical University-Plovdiv branch, Bulgaria [2]

valentina_nikolova@abv.bg [1]; expert2009@abv.bg [2]

Abstract: *The body of literature on business process compliance is large and approaches specifically addressing process monitoring are hard to identify. Moreover, proper means for the systematic comparison of these approaches are missing. Hence, it is unclear which approaches are suitable for particular scenarios. The goal of this paper is to define a framework for Compliance Monitoring Functionalities (CMF) that enables the systematic comparison of existing and new approaches for monitoring compliance rules over business processes during runtime. To define the scope of the framework, at first, related areas are identified and discussed. The CMFs are harvested based on a systematic literature review and five selected case studies. The appropriateness of the selection of CMFs is demonstrated in two ways: (a) a systematic comparison with pattern-based compliance approaches and (b) a classification of existing compliance monitoring approaches using the CMFs. Moreover, the application of the CMFs is showcased using three existing tools that are applied to two realistic data sets. Overall, the CMF framework provides powerful means to position existing and future compliance monitoring approaches.*

Keywords: BUSINESS PROCESS MANAGEMENT; BUSINESS PROCESS COMPLIANCE; COMPLIANCE MONITORING

1. Introduction

The need to check for compliance of business processes based on a set of constraints may emerge in different phases of the process life cycle [1],[2] ,[3]. During design time, the compliance of a process model with a set of constraints is checked. At runtime, the progress of a potentially large number of process instances is monitored to detect or even predict compliance violations. For this, typically, terms such as compliance monitoring or online auditing are used. Finally, processes can be diagnosed for compliance violations in a *post mortem* or offline manner, i.e., after process instance execution has been finished.

This paper is dedicated to compliance monitoring as this is crucial for the timely detection and prediction of compliance violations as well as for the provision of reactive and pro-active countermeasures on compliance violations [4], [5], [6]. Further, in realistic settings, the existence of a complete process model for compliance checks cannot always be assumed. In fact, business processes are often implemented in a rather implicit manner and executed over different information systems (e.g., Enterprise Resource Planning (ERP) or Customer Relationship Management (CRM) tools) as depicted in Fig.1. Although there are similarities between design time/*post mortem* analysis and compliance monitoring , this paper will focus on the latter in order to provide a clear scope.

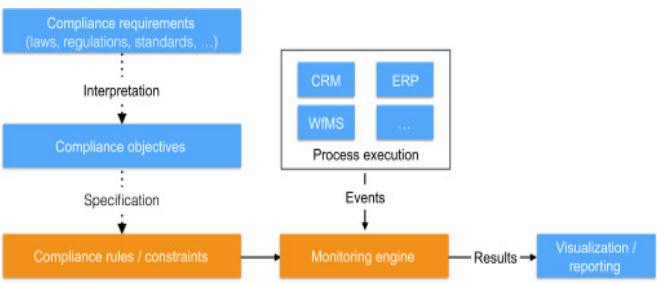

Fig.1 *Process model for compliance*

Typically, compliance requirements on business processes stem from different sources such as laws, regulations, or guidelines that are often available as textual descriptions. An important task towards compliance monitoring is the interpretation of these requirements as compliance objectives and the subsequent specification as compliance rules or constraints (note that, in this paper, we will use both terms interchangeably). As shown in Fig. 1, the specified compliance rules will be verified over the process execution events. The results of compliance monitoring can be visualized and reported back to users in different ways, ranging from notifications on violations to fine-grained feedback on reasons for violations, or even the prediction of possible and unavoidable future violations.

In general, compliance monitoring approaches are driven by two factors: (1) the *compliance rule language* that is used to specify the compliance requirements and (2) the *event format* the compliance checks are based on. Due to the possible heterogeneity of the data sources employed, an integrated target event format is desirable.

2. Research methodology

Hence, the main challenge tackled in this paper is to provide proper means for comparing approaches for compliance monitoring in business processes in a systematic way. This challenge will be addressed by the following four research questions: *How to identify compliance monitoring approaches?; What are functionalities that are essential for compliance `monitoring approaches in business processes?; How can we demonstrate the appropriateness of the identified compliance monitoring functionalities?;How can the compliance monitoring functionalities be applied in existing tools?*

The goal of this paper is to define a framework for Compliance Monitoring Functionalities (CMF) that enables the systematic comparison of existing and new approaches for monitoring compliance rules over business processes during runtime. Specific challenges for eliciting the CMFs are the multitude of existing approaches in the area of business process compliance and the decision of which functionalities are required in real-world scenarios. In order to address these challenges, we apply the methodology depicted in Fig. 2

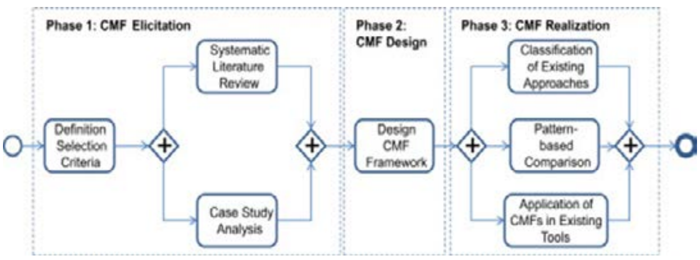

Fig.2: *Research methodology*

The methodology consists of three phases, i.e., *elicitation*, *design*, and *realization* of the CMF framework.

Phase1—Elicitation: The elicitation phase follows the research methodology described in the context of elicitation of process change and time patterns[7], [8], [9], [10]. First of all, *selection criteria* are defined that scope the research done in this paper. As overarching selection criteria, we focus on:

a. *functionalities that are relevant for process compliance monitoring*, i.e., the observation and enforcement of compliance constraints that are imposed over business processes during *runtime* and

b. *constraints that are imposed at the process level*, i.e., we exclude, for example, integrity constraints [6], [11], [10], [17].

The elicitation phase includes a systematic literature review and an analysis of five case studies from different domains. The CMF identification is based on the results of the systematic literature review and the case study analysis and possibly illustrated by additional examples.

Phase 2—Design: The CMF design itself is presented in *Section 4*. Each CMF is described using a CMF template and illustrated by examples taken from literature or case studies.

Phase 3—Realization: In order to demonstrate the appropriateness of the CMF design, existing approaches are classified along their support for the CMFs (cf. *Section 4*). Moreover, constraint patterns as suggested by the literature are compared to the CMFsproposed in this paper. Finally, to illustrate the application of the CMF framework, compliance rules are extracted from two realistic and publicly available data sets.

Based on the extracted rules and the data sets we showcase the application of compliance monitoring for business processes in existing tools (*cf. Section 5*). Note that the data sets for CMF application are different from the case studies utilized for CMF elicitation.

3.Compliance Monitoring Functionality (CMF) elicitation

After defining the selection criteria, the first input for the elicitation of CMFs is compiled from a systematic literature review.The search was conducted using scholar.google.com. In a first step, keywords were searched in the titles of the papers, excluding patents and citations. The list of references resulting from the horizontal search was evaluated in a first round excluding papers that: are clearly geared towards design time aspects; do not refer to requirements on the compliance specification language;do not refer to constraints at the business process level, but to more low level integrity constraints such as Service Level Agreements (SLAs) or calculating Key Performance Indicators (KPIs).

Obviously, the first filter criterion led us to loose design-time approaches that can be potentially lifted to runtime analysis. The results from the first and second round of the literature review were analyzed in two rounds; first of all, by assigning each paper to a researcher, followed by a group discussion on all 83 papers. For each paper, it was checked whether it *provides a compliance monitoring approach for business processes* [5],[8]; *includes studies on process compliance patterns* [9], [10], *provides enabling technologies and related techniques for process compliance monitoring, e.g., conformance checking* [11], [12], [13]; *provides frameworks for compliance monitoring infrastructure* [14], [15], [16] or *contract monitoring* [6], [7], or *features domain-specific approaches such as from health care, providing requirements, examples, and case studies* [13], [16].

4. Compliance Monitoring Functionality (CMF) design

This section presents our CMFF, i.e., the framework of *Compliance Monitoring Functionalities* (CMFs). Following the methodology set out in *Section 3*, we derived CMF candidates from a systematic literature review and five case studies. Based on several rounds of discussions, these candidates were then cleaned and aggregated into the ten CMFs proposed in this paper. Each CMF is described using the following template listing the *name*, a brief *overview* on the CMF, a *description*, guidelines about the *evaluation criteria*, *examples*, and *implementation* hints of compliance rules illustrating their functionality. Whenever possible we directly borrow the examples from literature or the projects. Sometimes we also provide new examples to highlight specific features of the CMFs.

Moreover, the following requirements for CMFs were identified that also serve as basis for classifying the ten presented CMFs . Such requirements tackle the three main dimensions of any CMFF: (a) modeling of compliance constraints, (b) analyzing the raw data at runtime, and (c) generating compliance monitoring results to be returned to the end users.

1. *Modeling requirements.* A compliance monitoring approach has to enable the specification of compliance constraints can be monitored The CMFs of this class refer to the ability of compliance monitoring approaches to express constraints not only on the control flow of a business process, but also on other, equally important, perspectives. This helps in classifying CMFFs with respect to their modeling capabilities, and to position their adequacy in a specific domain with its own compliance constraints to be formalized. The following three CMFs refer to the ability of a compliance monitoring approach to deal with constraints that address aspects beyond control flow*: time, data, and resources.*

2. *Execution requirements.* Compliance monitoring approaches should deal with execution-based information attached to the events of the event stream to be monitored. In general, an event is always related to an activity in a business process, but further information can be also provided. There are several requirements imposed on compliance monitoring approaches by the domain. The following CMFs enable the assessment whether or not a compliance monitoring approach meets these requirements: *Supporting non-atomic activities,Supporting activity life cycles; Supporting multiple instances constraints.*

3. *User requirements* The third dimensions focus on the ability to return the compliance assessment to the end users. Specifically, *advanced diagnostics and recommendations* relate to the ability of a CMFF to provide advanced, meaningful information to end users that go beyond violation detection and explanation. The CMFs described in this section refer to the ability of a compliance monitoring approach to address user requirements: *Ability to reactively detect and management; Ability to pro-actively detect and manage violations;Ability to explain the root cause of a violation; Ability to quantify the degree of compliance.*

5. Compliance Monitoring Functionality (CMF) realization

This section is concerned with *Phase 3 CMF Realization* of the methodology depicted in Fig. 2. Phase 3 consists of three building blocks, i.e., a pattern-based comparison of CMFs related to language aspects with compliance patterns set out in the literature, a classification of existing monitoring approaches using the CMF framework and the application of the CMFF in selected tools.

The authors of this paper proposed three approaches, namely *MobuconEC (cf. Section5.1), MobuconLTL (cf. Section5.2)* and *SeaFlows (cf. Section5.3)* , that make use of different techniques to enable compliance monitoring. In the following, we showcase the application of the ten CMFs in these three tools. As these are the tools we know best, it is ensured that we are able to correctly apply the tools in a case study for discussing the implementation of CMFs.

In order to analyze tools using the CMFs, a data set consisting of compliance rules covering the CMFs to be investigated and process instances (or process logs for replaying process instances) are necessary. We selected the Business Process Intelligence Challenge (BPIC).

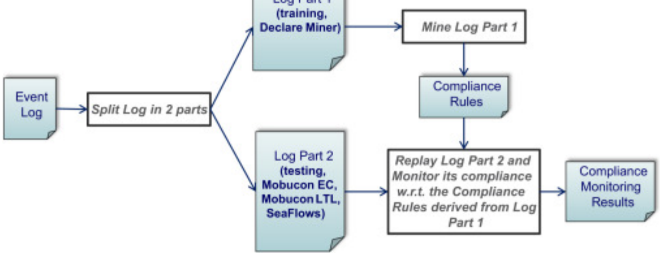

Fig.3.*Methodology of implementation of CMFs in the selected tools*

In this section, we illustrate the methodology we have followed to show how the CMFs have been implemented in the selected tools. We started from the data sets described in Fig. 3. illustrates the methodology. We have split the logs in two parts (Log Part 1 and Log Part 2). For the hospital log, in the first part, the first 571 cases are considered. In the second part, the rest of the cases (572 cases) are included. For the financial institute log, in the first part, the first 6543 cases and, in the second part, the rest of the cases (6544 cases) were considered. Then, we have mined Log Part 1 (training log) to extract a set of compliance rules. To do this, we have used the *Declare Miner* component of the process-mining tool ProM [1], [2], [17]. This allows us to automatically discover compliance rules related to control-flow, data and resources based on the mainstream observed behavior, i.e., frequent behavior is converted into a collection of *Declare* constraints. *Declare* is a declarative language based on an extensible set of constraints [12], [13], [14] , [15]. *Declare* supports most of the modeling constructs and hence a good candidate to evaluate and illustrate the CFMs. The compliance rules have been used as a reference model to monitor a stream of events coming from the replay of Log Part 2 (testing log).

5.1. Compliance monitoring with MobuconEC

MobuconEC is a compliance-monitoring framework based on a reactive version [12] of the Event Calculus [13]. As described in [2] and [6], the approach has been exploited to formalize the extension of Declare described in [3] and [4]. This extensions support: • Metric time constraints, which can be directly formalized (together with qualitative time constraints) using the *explicit approach* to time provided by the Event Calculus ; •Data and data-aware conditions, leveraging on the first-order nature of the Event Calculus ;• Resources, which are considered as special data;• Non-atomic activities, which can be encoded in the Event Calculus using additional data and dedicated rules, as shown in [7] .Notably, the formalization in [1] directly support the possibility of monitoring the lifecycle of activitie.The high expressiveness of compliance rules in *MobuconEC* has the main drawback that only reactive management of violations can be tackled: proactive management would require to reason on the possible future continuations of the current, partial trace by considering also data and metric timestamps, which is undecidable.

5.2. Compliance Monitoring with MobuconLTL

MobuconLTL is a compliance-monitoring tool implemented as a provider of the operational support in the process-mining tool ProM. It takes as input a reference model expressed in the form of Declare rules. More generally, every business constraint that can be expressed as an LTL formula can be monitored using *MobuconLTL*. A stream of events encoded using XES can be monitored with respect to the given LTL specification.

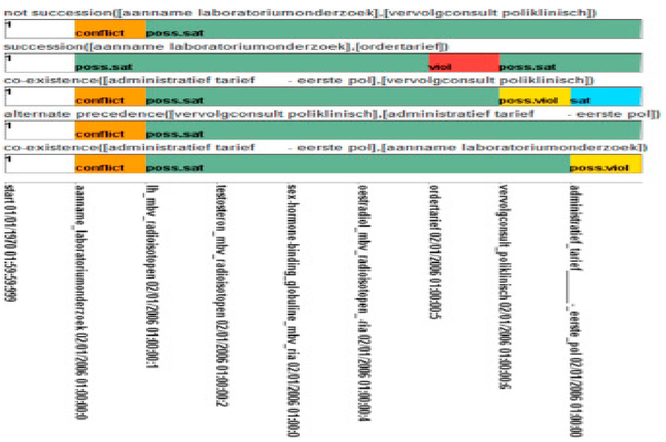

Fig.4. *MobuconLTL supports metrics for quantifying the degree of compliance*

MobuconLTL monitors finite-trace LTL constraints through deterministic finite state automata. Therefore, it does not tackle constraints referring to data and resources (ranging over finite state) because of the state space explosion problem. *MobuconLTL* implements reactive monitoring . It is possible, at runtime, to detect that a state of affairs is reached such that two or more compliance rules become conflicting (conflict state); the presence of a conflict means that no possible future course of execution exists such that all the involved constraints are satisfied. In this sense, *MobuconLTL* also supports a pro-active management of violations and root cause detection. The automata-based approach allows *MobuconLTL* to provide the user with detailed diagnostics about which activities can be executed and which ones are forbidden at any point in time during the process execution. This is possible by evaluating which transitions can be fired from the current state of the automaton and which ones bring the automaton in an inconsistent state. The tool only supports simple metrics for quantifying the degree of compliance of a case (Fig.4.).

5.3. Compliance monitoring with SeaFlows

SeaFlows is a compliance checking framework that addresses design and runtime checking. It aims at encoding compliance states in an easily interpretable manner to provide advanced compliance diagnosis. The core concepts described in [3] , [4] [5], [6] were implemented within the prototype named *SeaFlows Toolset*. With *SeaFlows*, compliance rules are modeled as *compliance rule graphs* (CRG). *SeaFlows* enables to monitor a stream of events encoded in a predefined event format. It further enables the import of logs in XES standard format.

Qualitative time constraints are well-supported by *SeaFlows*. In particular, the CRG approach is not restricted to predefined compliance constraint patterns but allows for constraints that are more complex. However, *SeaFlows* does not address quantitative time .

SeaFlows only partially supports CMF 2. This is because it provides only limited support for constraints with non-unary data conditions. Resource-aware compliance rules are only supported if the resource conditions can be expresse

For this case study, the logs were automatically replayed and checked against all modeled rules. For each violation, a violation file was created. Fig.5 illustrates how compliance with R5 is monitored by replaying the log of a specific case.

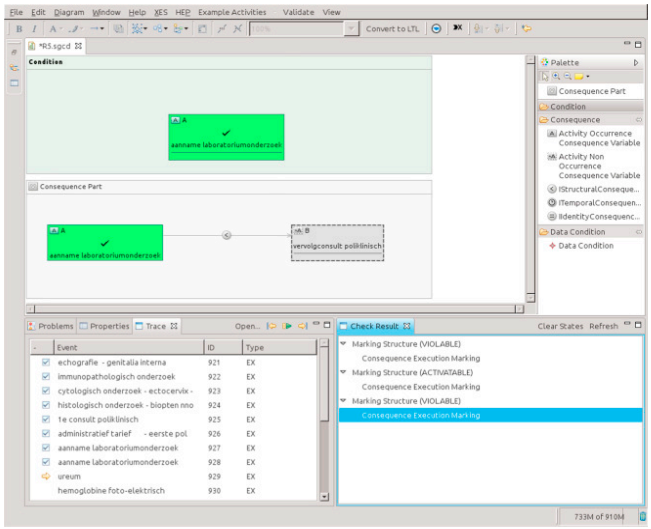

Fig.5. *SeaFlows monitoring of compliance by replaying the log*

As compliance with a CRG is checked by executing it against the event trace using well-defined rules, the compliance state is represented through markings of the CRG. When a violation is observed, it is reflected in the markings of the CRG.

When the last event of the case is executed, both instances become violated as shown in Fig. 6. The marked CRG serves as basis for deriving explanations for violations. That is the reason why the corresponding node is marked with red color. Once violated, the instance of a compliance rule cannot become satisfied in the further process execution. This is because the violation is already manifest in the log. However, monitoring can still be continued for the compliance rule (i.e., future possible violations of the rule in the process instance can also be detected). Thus, *SeaFlows* supports reactive compliance management.

As the compliance monitor of *SeaFlows* was implemented as a proof-of-concept for the reporting features for compliance monitoring. Although the compliance states of multiple compliance rule instances can be aggregated to provide an overall compliance level, the framework does not support a more detailed analysis of the individual metrics. In addition, *SeaFlows* does not detect violations caused by the interplay of two or more constraints. Hence, the conflicts among some of the rules in the case study remain undetected.

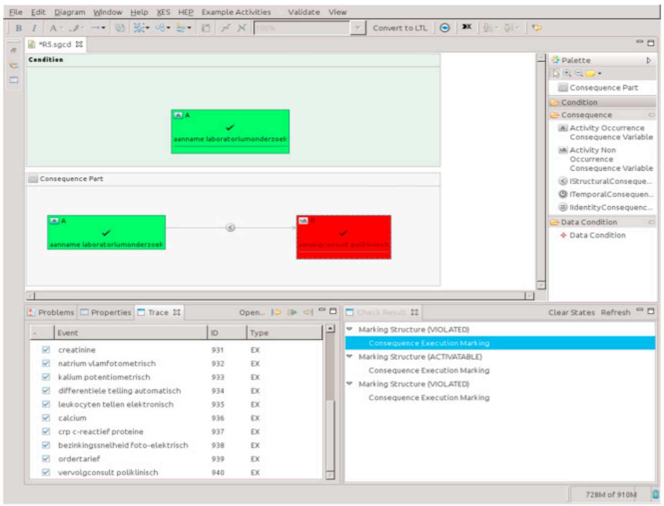

Fig.6. *SeaFlows monitoring of compliance by replaying the log*

6. Conclusion

This paper presents a framework for the systematic comparison of compliance monitoring approaches in the business process management area. The framework consists of *ten Compliance Monitoring Functionalities* (CMFs) and includes requirements for the constraint modeling notation (e.g., supporting time and data), requirements with respect to the execution (e.g., supporting multiple constraint instances), and user requirements (e.g., providing fine-grained feedback). The CMFs are harvested based on a systematic literature review as well as from five case studies from different domains (health care, food and diary manufacturing and maritime safety).

The appropriateness of the CMFF is shown in two ways. First of all, the CMF framework is compared with existing compliance patterns in the business process management area. Secondly, existing compliance monitoring approaches are classified based on their support for the CMFs. The comparison with compliance patterns supports the importance of the four constraint-related CMFs, i.e., those CMFs that relate to language and expressiveness aspects in the constraint specification. The classification of existing approaches pointed out that none of them supports more than seven CMFs and most approaches are not supported by publicly available software tools.

Here we have to note that for some approaches several of the CMFs could not be evaluated based on the literature. It seems that several approaches focus on a specific language aspect rather than integral monitoring support. Less attention has been devoted to user requirements, i.e., the provision of fine-grained feedback or even the proactive indication of compliance violations. Nevertheless, it is crystal clear that users play an important role in compliance monitoring, e.g., to interpret deviations.

In order to demonstrate the applicability of this CMF framework, two realistic data sets from the BPIC 2014 and 2015 were applied using three compliance monitoring tools, i.e., *MobuconLTL*, *MobuconEC*, and *SeaFlows*. The data sets consisting of process execution logs from patient treatment and the financial domain were divided into training and testing data sets. Compliance constraints were harvested based on the training set, implemented within the three tools, and monitored over the testing data set. The application of concrete tools to these data sets nicely illustrates how compliance monitoring works in practice and shows what the interaction with users looks like.

The work can be further extended in several directions, e.g., to cross-organizational or configurable processes. An interesting area of adaptation/extension of the CMF framework here presented naturally arises when there is the need of a comprehensive compliance evaluation not just within, but also across process cases. Traditionally, business processes are monitored in a case-by-case manner. However, compliance rules may span across cases, e.g., because they focus on resources independently from the specific case in which they operate, or because they need to compare and combine data produced inside different cases.

References

[1] S. Sadiq, G. Governatori, K. Naimiri, **Modeling control objectives for business process compliance,** in: International Conference on Business Process Management,(2007), pp. 149–164

[2] L.T. Ly, S. Rinderle-Ma, K. Göser, P. Dadam **On enabling integrated process compliance with semantic constraints in process management systems** Inf. Syst. Front., (2012), pp. 195–2

[3] S. Sadiq, **A roadmap for research in business process compliance**, Business Information Systems Workshops, (2011)

[4] F.Caron, J. Vanthienen, B. Baesens, **Comprehensive rule-based compliance checking and risk management with process mining**, Decis. Support Syst., 54 (3) (2013), pp. 1357–136

[5] M Kharbili, S. Stein, I. Markovic, **Towards a framework for semantic business process compliance management,** in: International Workshop on Governance, Risk and Compliance—Applications in Information Systems (GRCIS′08), (2008), pp. 1–15

[6] Gigova T. **Business process modeling in industrial enterprises**, 12th International Conference "Research and Development in Mechanical Industry", Serbia, (2012), pp. 431-435

[7] M.Weske, **Business Process Management: Concepts, Languages, Architectures,** Springer, Berlin, Heidelberg, New York (2007)

[8] Alexieva V.**,"BPM model at the enterpize level ",** Journal PROCEDIA – Social and Behavior, Publ. by ELSEVIER, (2013)

[9] Alexieva V.,**Process management and adequacy of the Bulgarian business in a turbulent environment,** International scientific conference "Economics and Management in turbulent business environment,"Sofia, (2014)

[10] Mihova T., Alexieva V., Gigova T., **Risk management in industrial enterprises,** International virtual journal for science, technics and innovations for the industry**,** Issue 9 /2014

[11] Mihova T.,**How effective is the monitoring of projects for small business in Bulgaria?**,International scientific conference "Project Management", Tryavna,(2011)

[12] Dinkov Hr., N. Bozukov, Iv. Gaytandzhiev**,** M. Dimitrov, **Advantages of measurement systems based on digital sensors, estimating the real effect of an energy saving project application,**"International Conference on Application of Information Technology and Communications and Statistics in Economy and Education",Sofia, Bulgaria, (2012), pp. 417 - 420

[13] Bozukov N., M. Dimitrov, Hr. Dinkov**, Data and methods for measurement of energy savings,** International Conference on Application of Information Technology and Communications and Statistics in Economy**,** Sofia, Bulgaria, (2012), pp 421 - 423

[14] Georgieva, S., P. Yordanova-Dinova**, Opportunities linear programming in drawing up the production program of the enterprise in the food industry,** X Jubilee national scientific conference"Ecology and Health", Plovdiv, Bulgaria, (2014)

[15] Yordanova-Dinova, **Trends in working capital of мeat processing enterprises (the example of a sub-region of Plovdiv 2006-2010) in the context of inter-company indebtedness,** Scientific Works "Food Science, Engineering and Technology, Volume LIX UFT-Plovdiv, (2012)

[16] Yordanova-Dinova, S. Georgieva, **Inter comparative analysis of commercial performance to the food industry,** Scientific Works of the UFT - Plovdiv, vol. LXI,Bulgaria, (2014)

[17] Alexieva V., **Survey of how process modeling is used in the Bulgarian organizations,** Journal: PROCEDIA – social and behavioral science, Published by ELSEVIER,(2013)