The Future of Gas Turbine Technology
8th International Gas Turbine Conference
12-13 October 2016, Brussels, Belgium

Paper ID Number (52_IGTC16)

A DIAGNOSTIC & CORRECTIVE ACTION SYSTEM BASED ON DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

Dr. Thomas Hubauer, Dr. Mikhail Roshchin, Dr. Giuseppe Fabio Ceschini, Almir Avdovic, Siemens AG

name.surname@siemens.com

ABSTRACT

This paper presents a novel approach to support engineers in the Remote Diagnostic Centers developed at Siemens. An innovative combination of Deep Learning[1] and Natural Language Processing (NLP)[2] technologies allows us to harness both the vast amount of the engineers' diagnostic knowledge as well as information about the current turbine status derived from the available sensors. This approach is embedded into an overall systematic workflow building on physics-based, rule-based and data-driven methods. Based on results from a recent R&D project at Siemens Remote Diagnostics Services, we prove that this framework successfully supports engineers in identifying relevant information, thereby significantly reducing trouble shooting time and increasing both technical responsiveness capability and capacity.

INTRODUCTION

Both power generation and oil and gas domains operate turbo-machinery plants using complex rotating equipment. To keep up with the technological advances, the turbo-machinery industry aims to integrate engineering, manufacturing, servicing and maintenance of their plants into a single technological eco-system. The classical approaches are the utilization of condition monitoring services and diagnostic solutions, as well as the integration of design models and simulation results into the plant operations. This results in better availability and reliability of plant systems, improved operations, lower maintenance cost, and higher safety.

Remote monitoring and diagnostics of rotating equipment provided directly by the original equipment manufacturer (OEM) are indispensable in practice, and a focus of active research in academia. Recently, there has been an increased demand for a systematic approach to plant process safety, increased reliability and availability, lower maintenance cost, and continuous awareness about the equipment health status. This poses a challenge to the existing tool landscape, which typically relies on the adaptation of condition monitoring solutions to expert systems. Specifically, fault detection, fault isolation, failure mechanism definition and diagnosis definition as part of the systematic diagnostics are essential to support OEM engineers in their decision making process, up to and including the corrective action recommendation. However, due to the technical complexity caused by the large number of subsystems and process flows, diagnosis for industrial gas turbines is non-trivial, and requires the multi-disciplinary expertise of various engineers from domains such as system mechanics, aerodynamics, and thermodynamics, to name but a few.

Specifically, root-cause analysis as part of failure analysis within system engineering & diagnostics is an indispensable feature of the design and maintenance phase. It allows identification of faults based on their causes and effects that propagate at different system levels. Consequently, a model-based approach to failure analysis for industrial gas turbine applications is used to realize an efficient system analysis. Over the last decades, a natural evolution of the systematic approach can be observed, which can be summarized as an extension of the original (1) physics-based methods by (2) rule-based systems[3], and over the last years additionally utilization of the (3) data-

[1] Deep Learning denotes a class of machine learning methods that use multiple model layers to discover high-level abstractions of the input data and use it for analytical tasks. Current approaches are often based on the neural network paradigm.

[2] Natural Language Processing is a field of Artificial Intelligence focusing on the processing of human language by computers. For this paper, we focus on the aspect of Natural Language Understanding, i.e. the interpretation of human-generated content (here: text) by computers.

[3] Systems utilizing (typically hand-crafted) IF-THEN rules to derive new information from known facts.

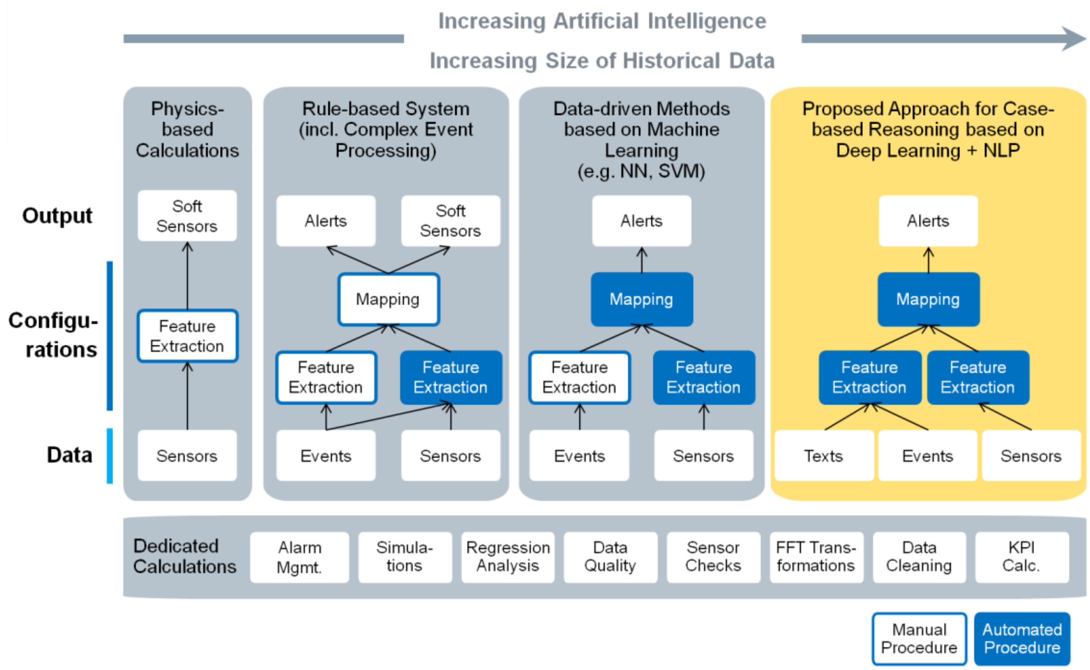

Figure 1: Knowledge types and processing methods for efficient remote diagnostics

driven methods[4] for many-dimensional operational characteristics. This three step approach based on physics, rules and machine learning enables the formalization of various qualitative models of key turbine components, which are highly error-prone, together with their potential failure mode descriptions, and their impact at different system levels. This information is collected over decades in a single, large knowledge base and regularly processed (semi-)automatically using various available reasoning engines.

Thus, there is an urgent need to utilize all available historical operational data of the rotating equipment to leverage the hidden knowledge. In order to make this feasible we need to address all aspects of the problem: (1) sensor data from the turbine instrumentation, (2) sequences of events from control systems, complemented by (3) reports (i.e. textual information) which can be understood as annotations of the raw data. This approach constitutes an extension of (already) existing hybrid diagnostic systems in order to fulfill current requirements in both failure detection and isolation for industrial systems.

STATE OF THE ART

As of today, diagnostic systems typically make use of at least four types of knowledge (see Figure 1):

1. Dedicated calculations for simple data analysis tasks including, but not limited to: data quality and sensor checks, data cleaning, transformations such as FFT, KPI calculations, prediction, alarm management, and simulations.
2. Engineering knowledge about physics-related processes formalized on the basis of simulations performed during design and manufacturing (e.g. thermo-dynamical calculations for performance monitoring).
3. Operational experience about incorrect or non-desired behavior under certain scenarios formalized as IF-THEN rules (i.e. fault models, either declarative or hard coded, including so-called complex event processing, or CEP[5]).
4. Baseline models that describe the normal state of the system and allow identification of deviations, typically formalized with the help of various machine learning methods such as Neural Networks and Support Vector Machines[6].

In addition, there have been various attempts to utilize case-based reasoning (Kolodner, 2014), providing a similarity function over the multi-dimensional space of sensor data.

However, all such systems fail to include the experience stored in a semi-structured format, for instance in experience reports or service tickets, and to make it available for automated decision support. Such experience

[4] Methods for automatically discovering dependencies from historical data, often also called machine learning.

[5] CEP is an extension of rule-based systems taking time into account. This allows detecting temporal pattern characterizing, for instance, harmful processes.

[6] Neural Networks (NNs) and Support Vector Machines (SVMs) are two examples of so-called supervised machine learning methods. Based on large sets of historical data together with so-called labels (that tell the algorithm whether the corresponding data denotes a normal or deviant state), such models allow to automatically classify new data into these groups to assess the current status.

summarizes past situations with their characteristics, decisions taken, and (ideally) documented effects of the related actions. Such information is very valuable for several reasons: Firstly, it often comes “for free” as an artifact generated, for instance, in the maintenance process. Secondly, experts often find it easier to express their experience in natural language than in strictly deterministic rules. And thirdly, while data-driven approaches often need to be adapted to each specific device, the information expressed in such texts is generally more abstract, therefore easing generalization from one device to others of the same family. The proposed in the paper approach to decision support for remote diagnostics of gas turbines, which takes into account both operational and human-generated content, can be seen as a specific type of case-based reasoning (Korbicz et al. 2002) based on a combination of Deep Learning and Natural Language Processing technologies.

RELATED WORK

In the context of rotating equipment engineering, gas turbines move products such as gas for power generation or mechanical drive applications. In general, every unit of a plant consists of a driven machine, a driver, and a transmission device and is supported by auxiliary equipment as discussed by Forsthoffer (2011). Figure 2 shows the topology of an industrial plant.

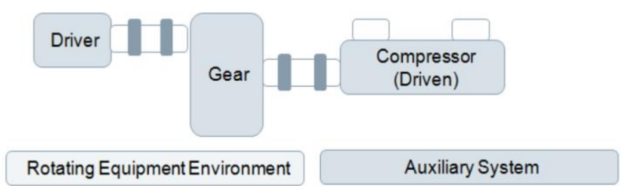

Figure 2: Industrial plant landscape

All equipment mentioned above can be classified further and can have different configurations. For example: drivers can be classified as steam turbines, gas turbines, motors (induction, synchronous or vari-speed) or engines (internal combustion, diesel or gas). The key is to understand the functionality of the equipment’s critical components in order to effectively monitor and maximize plant safety and reliability. Reliability is commonly defined as the operating time of the equipment per year. It is a measure of the equipment unit performing its specified function without a forced (unscheduled) outage for a given period of time (Forsthoffer, 2011). In case of an outage, the loss of revenue can exceed one million U.S. Dollars a day according to Forsthoffer (2011). An outage is usually caused by the shutdown of a critical component. Many leading companies including our industrial partners recognize the critical importance of reliability management and adopt the following strategies (Ceschini and Saccardi, 2002):

1. Involve the end-user in the specification, design and installation phase of the plant;
2. Determine the life span of the plant and its components, which is extremely long compared to the development phase;
3. Analyze the instrumentation and location of the plant since these directly impact the equipment’s reliability;
4. Focus on the design and installation due to the substantial influence on the maintenance requirements, its cost and the availability of a particular piece of machinery.

One of the best systematic overviews of all crucial diagnostic components of the overall ecosystem is given by Korbicz et al. (2002). In addition, an extensive summary of the existing techniques and methods for industrial diagnosis is provided in Vachtsevanos et al. (2006). Both suggest the usage of a systematic approach taking into account the nature of the monitored system, interrelations of the subsystems and their components, instruments and sensors, as well as all available engineering tools and various methods. There is no silver bullet in mathematics or machine learning to address every small aspect with one method. Hence the overall diagnostic ecosystem will be always based on hybrid approach, involving five major steps: (1) basic calculations on available system parameters, (2) physics-based calculations of the relevant processes of a system, (3) rule-based methods to support a knowledge-driven approach, (4) data-driven techniques to find complex correlations in the data, and (5) case-based reasoning to adapt existing solutions to known issues from the past.

Points (3) and (4) of this list are described to a certain extent in the papers by Mehdi et al. (2015), Hubauer et. al. (2013) and De Haan & Roshchin (2012). They identify the current issues of the standard approach and propose extensions to the diagnostic landscape at the Remote Diagnostic Center at Siemens.

PROPOSED APPROACH

In contrast to the solutions presented in the preceding section, our novel approach combines machine-generated and human-generated content in order to support engineers in the Siemens Power Generation Remote Diagnostics Center (RDC). More concretely, our solution utilizes similarity-based reasoning over textual case descriptions, sensor data and sequence-of-events (SoE) data to present the technician with an ordered list of most similar historical cases. By inspecting these cases, the engineer is quickly guided to a solution for his problem. Figure 3 depicts the conceptual approach of our solution, with input / output depicted in grey and processing steps in white. In the following sections, we provide insight into the processing flows for machine data as well as for human-generated content, and show how the information extracted from both sources is combined into a single, consistent answer presented to the end user.

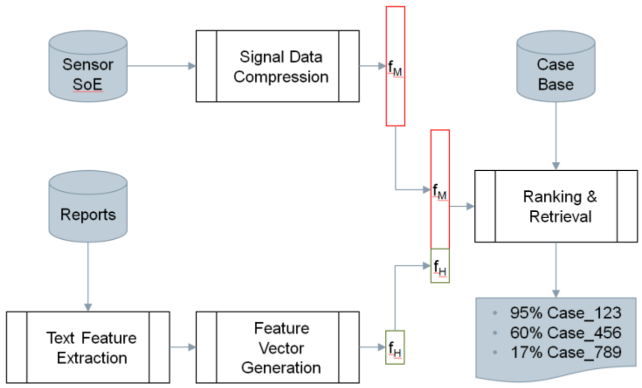

Figure 3: Functional overview of the proposed solution

Processing of human-generated content

In our solution, the maintenance tickets created (and updated) by a remote service engineer during the analysis of a new case constitute one of the major sources of information. This data contains at least the following information:

- The unique identifier of the affected turbine,
- a short title of the case, summarizing the problem at hand in a few words (in English language),
- a longer case description detailing the findings and interpretations of the engineer(s) (also in English language),
- information on the date and time the problem occurred (typically close to the creation date of the ticket).

The textual information is processed by an automated feature extraction pipeline. In a nutshell, we first extract relevant keywords from the texts, which are in turn normalized by applying standard natural language processing, or NLP (Winograd, 1972) technologies such as stemming. In a second step, our pipeline extracts component-symptom relationships from the text. This is done by means of a domain-specific grammar, which has been developed jointly by NLP and subject matter experts. By using a grammar-based approach in addition to established keyword-based methods, our solution is robust to misspellings and alternative formulations since it captures the inherent semantics of the text in addition to its written, symbolic representation.

The numeric NLP feature vector of a case (denoted in the figure by f_H) is then derived by probabilistic assignment of the case to a pre-trained cluster model, i.e. entry *i* of the feature vector is the probability of the case belonging to cluster *i*. The dimensionality of the NLP feature vector is therefore given by the number of clusters of the model, currently set to 50. This cluster model has been trained based on a corpus of historic cases processed by our feature extraction pipeline. In an iterative process involving feedback by domain experts, we have optimized this model to yield clusters with consistent underlying causes. This way, the experts' notion of similarity of cases is captured in the clustering model and applied to new cases automatically.

Processing of machine data

Processing human-generated content as presented in the previous section elegantly integrates expert knowledge and human intuition into the analysis process. However, there are a number of reasons for us not to rely solely on this information source:

1. We want to provide our support functionality as early as possible in the diagnostic process, without requiring the user to do intensive analyses as a prerequisite.
2. We want the tool to also provide support in cases of hitherto unseen failure classes (or unforeseen ways of describing a situation) for which no conclusive clusters exist.
3. Diagnostic engineers new to the job (or a specific family of turbines) shall be supported even if they are confronted with cases they would not know how to tackle otherwise.

Therefore, we integrated a second processing pipeline that works directly on the sensor and sequence-of-events (SoE) data provided by the control system of the turbine. This is exactly the same data that a human engineer would inspect when going through the diagnostic process without any tool support. This part of our solution can therefore be understood to mimic the analysis process of the expert. Different from a human expert, however, larger amounts of data (i.e. more sensors, more events, and/ or larger time slices) can be taken into account due to the sheer computational power available.

Based on the feedback from subject matter experts, we currently use data from the 24 hours prior to the incident for this analysis (the time being taken from the ticket contents described previously). To compute features and compress those 24 hours of data into the machine data feature vector f_M, we developed a method that combines event counts and sensor values over the time domain by means of so-called deep learning technologies (more concretely, a deep convolutional neural network, or CNN, trained over historical data). The internal layers of this network can be understood to conduct a stepwise compression of the information taking into account the temporal ordering of sensor and event data, as depicted graphically in Figure 4. The result of this step is a vector of length 1000. For more information on CNNs and Deep Learning, the interested reader is referred to (Krizhevsky, Sutskever and Hinton. 2012), (LeCun and Bengio, 1995), and (Sainath, 2013).

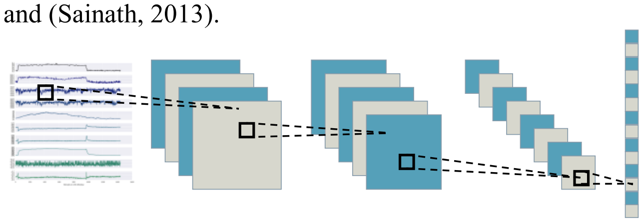
Figure 4: CNN approach for feature vector generation

Ranking and case retrieval

As shown in the right part of Figure 3, the two feature vectors f_H and f_M computed in the steps described above provide the basis for the ranking and retrieval of the most relevant historic cases. This ranking is done by computing the cosine-similarity of the (combined feature vector of the) newly created case to (the combined feature vector of) each historic case. To maximize flexibility with respect to the dimensionality of the two feature vector components, we first compute the similarity of the two components independently, giving us a "text similarity" and a "machine data similarity" value for each pair of cases. We then combine these two values using a machine learning model trained on the historic cases, resulting in a single similarity value (ranging between 0.0 and 1.0) for each pair of cases. The 100 historic cases most similar to the new case entered by the user are then shown to the diagnostic engineer in an ordered list, with a graphical indication of the degree of similarity.

REALIZATION AND EVALUATION RESULTS

We have implemented the proposed solution in a tool called "ADS" (Advanced Diagnostic System) which offers the above-described functionality to remote diagnostics engineers via a web interface. Our implementation is explicitly designed for an incremental diagnostics approach: The user enters initial findings into the case description field, gets an initial list of results, uses information taken from these cases to conduct deeper analyses and to update his case, issues the query again, etc – until he arrives at an acceptable solution. The web-based graphical user interface is depicted in Figure 5.

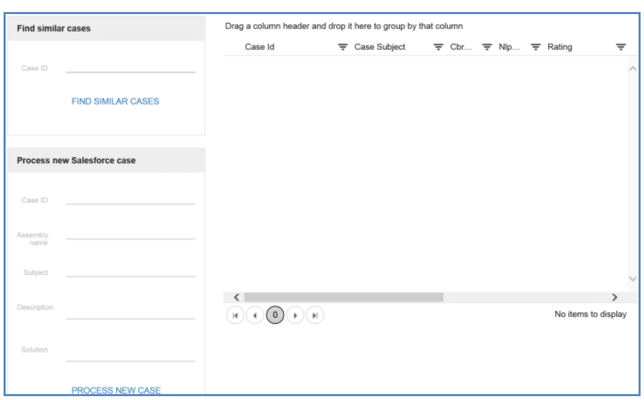

Figure 5: UI of the ADS system

Early during implementation of the developed system, we conducted an evaluation with selected key users as follows: For a total of 28 test cases chosen randomly from the historic case base, we provided the experts with the top 10 answers returned by the current version of our solution. The experts then judged the number of solutions in the list, and the number of "helpful answers" defined to be either a solution or guiding to find one. The evaluation results shown in Figure 6 highlight the outstanding potential of the ADS tool, which returned a minimum of four correct solutions for all test cases. Moreover, in over 50% of the evaluated cases at least 7 out of the 10 proposed answers were considered relevant by experts.

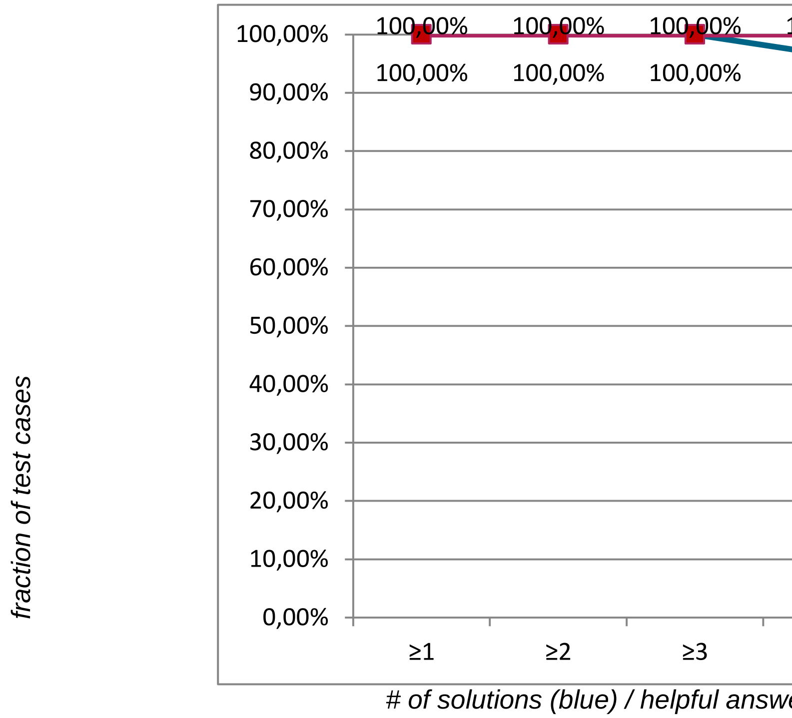

Figure 6: Preliminary evaluation results

CONCLUSION AND FUTURE WORK

We conclude the preliminary evaluation of our approach by emphasizing that the deep learning model including sensor data had not yet finished training at the time of testing. However, even the reduced approach based on textual case descriptions and SoE data yielded high quality results; the more powerful approach including sensor data is expected to result in a further major improvement of the quality of the proposed cases. We are currently working on an updated evaluation to quantify the additional quality boost.

Recently, the growth of computational power has given new impetus to autonomous decision making methods from the area of artificial intelligence. These developments made available new methods and tools to tackle the challenges outlined at the beginning of this paper. We have presented such a method: a novel Case Based Reasoning method combining Natural Language Processing and Deep Learning that provides high-quality recommendations in an interactive diagnostics scenario. The unique combination of machine data and human-generated content provides engineers in a remote diagnostic center with highly relevant (at least 4 solutions out of 10 suggestions) historic cases, given a new issue.

In the future, our solution will multiply the number of cases an engineer is able to handle in a day, thereby increasing efficiency of remote services tremendously. To this end, we are currently working on integrating the solution into the engineers' standard tools. In parallel, we plan to investigate more closely the aspects of sensor data and the effect of integrating it into the recommendation process as well.

REFERENCES

1. Ceschini G. F, Saccardi D. (2002). Availability centered maintenance (ACM), an integrated approach. In *Reliability and Maintainability, 2002 Annual Symposium on* (pp. 26-31). IEEE.
2. de Haan R., Roshchin M. (2012) Detecting Temporally Related Arithmetical Patterns – An Extension of Complex Event Processing. In *Knowledge Discovery and Information Retrieval (KDIR), 2012 International Conference on* (pp. 329-332). SciTePress.
3. Forsthoffer, W. E. (2011). *Forsthoffer's Best Practice Handbook for Rotating Machinery*. Elsevier.
4. Hubauer T., Lamparter S., Roshchin M., Solomakhina N., Watson S. (2013). Analysis of data quality issues in real-world industrial data. In *Prognostics and Health Management Society (PHM), 2013 Annual Conference of the*. phmsociety.
5. Korbicz J., Koscielny J. M., Kowalczuk Z., Cholewa W. (2002). *Fault Diagnosis – Models, Artificial Intelligence, Applications*. Springer Verlag
6. Kolodner, J. (2014). *Case-based reasoning*. Morgan Kaufmann.
7. Krizhevsky A., Sutskever I., Hinton G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS), Proceedings of the 25th Conference on* (pp. 2097-1005), Curran Associates, Inc.
8. LeCun, Y., Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10), 1995.
9. Mehdi G., Naderi D., Ceschini G. F., Roshchin M. (2015): Model-based Reasoning Approach for Automated Failure Analysis: An Industrial Gas Turbine Application. In *Prognostics and Health Management Society (PHM), 2015 Annual Conference of the*. phmsociety.
10. Sainath, T. N. (2013). Deep convolutional neural networks for LVCSR. In *Acoustics, Speech and Signal Processing, 2013 IEEE International Conference on* (pp. 8614 - 8618). IEEE.
11. Vachtsevanos G., Lewis F., Roemer M., Hess A., Wu B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. John Wiley & Sons Inc.
12. Winograd, T. (1972). Understanding natural language. *Cognitive psychology*, 3(1), 1972.