ISBN: 978-1-61208-154-0
DOI: 10.13140/2.1.3124.0647

An Artificial Intelligence Approach Towards Sensorial Materials

Florian Pantke, Stefan Bosse, Michael Lawo, Dirk Lehmhus
ISIS Sensorial Materials Scientific Center
University of Bremen
Am Fallturm 1, 28359 Bremen, Germany
{fpantke, sbosse, mlawo}@informatik.uni-bremen.de,
dirk.lehmhus@uni-bremen.de

Matthias Busse
Fraunhofer IFAM
Wiener Strasse 12, 28359 Bremen, Germany
busse@ifam.fraunhofer.de

***Abstract*—Sensorization aims at equipping technical structures with an analog of a nervous system by providing a network of sensors and communication facilities that link them. The objective is that, instead of having been designed to loads and tested to conditions, a structure can experience and report design constraint violations by means of real-time self-monitoring. Specialized electronic components and computational algorithms are needed to derive meaning from the combined signals. For this task, artificial intelligence approaches constantly gain importance; the more so as the trend of ever increasing sensor network size and density suggests that sensor and structure may soon become one, forming a sensorial material. Current simulation techniques capture many aspects of sensor networks and structures. For decision making and communication, the Intelligent Agent paradigm is an accepted approach, as is finite element analysis for structural behavior. To gain knowledge how sensorial structures can most effectively be built, an artificial intelligence based process for the construction of such structures was developed that uses machine learning methods for fast load inference. It is presented in this paper, along with evaluation results obtained in experiments using a finite element model of a strain gauge equipped plate which demonstrate the general practicability.**

***Keywords*-sensorial material; finite element method; sensor network; machine learning; multi-agent system**

I. Introduction

Adopting principles from nature, sensorization aims at equipping technical structures with an analog of a nervous system by providing a network of sensors and communication facilities that link them. Specialized electronic components as well as suitable computational algorithms are needed to derive meaningful information from the sensor signals. The main objective is to build structures that—instead of being designed once and tested for health based on externally observable exceptional events or in predefined intervals—constantly monitor themselves during operation by means of sensors which are attached to the structure's surface, directly printed on it [1], or embedded inside the material [2]. Such structures will be able to infer facts about their current loading state, e.g., autonomously detect overloading, as well as permanently record and carry with them their entire mechanical and/or thermal loading history. In order to attain high sensor densities, i.e., to embed hundreds or thousands of sensors into one single structure, all sensors and electronic circuits need to be miniaturized to a maximum degree. As these components usually have no significant load-bearing capabilities, the volume they consume within the structure directly affects the latter's mechanical stability. Thus, in the sensor integration process, avoiding the introduction of evident macrostructural defects is imperative.

An additional requirement, particularly in mobile applications, is a low power consumption profile of the electronic components. It simply is not feasible in most cases to equip the structure with a high-performance microprocessor unit along with sufficient RAM that would be necessary for inferring facts about the applied load cases from the sensor signals by employing the finite element method (FEM) during operation, for instance. Less computation- and resource-intensive approaches are called for. Especially if smart structures reacting to stimuli from their environment [3] are desired, a spatial distribution of numerous simple, miniaturized, low-power evaluation units over the entire structure seems advisable. As is the case for a human accidentally touching a hot plate, fast locally confined reactions preventing (further) damage may be vital for an engineered structure, while calculating, e.g., the millimeter-precise location of load introduction along with the global geometric state of the entire structure may be an unaffordable luxury. For long-term availability, the respective sensor nodes should ideally draw on local energy sources [4].

This article presents an approach to building sensorial structures utilizing basic methods from the field of artificial intelligence (AI). The central aim is to derive useful information despite having to cope with limited computational power and noisy sensor signals. Section II provides an overview of the basic AI approach towards building sensorial structures. Section III introduces a functional mockup system which will serve to implement and evaluate different approaches to sensor data assessment and load case inference. Section IV presents a robust sensor network developed for sensor data assessment, while Section V gives an overview of two simple machine learning methods that were used for

ISBN: 978-1-61208-154-0 Proc. of the Future Computing Conf. 2011
DOI: 10.13140/2.1.3124.0647

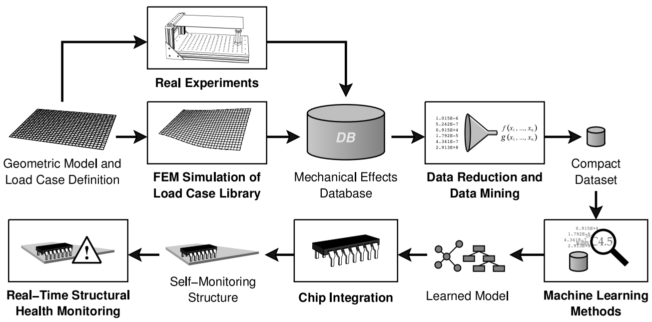

Figure 1. Artificial intelligence approach for the construction of sensorial structures

first experiments using strain data from FEM simulation. The evaluation results are presented in Section VI. Section VII portrays how the Intelligent Agent paradigm can be employed for monitoring more complex structures in a distributed fashion. Section VIII closes with a conclusion and outlook.

II. An AI Approach towards Sensorial Structures

The AI-based process model for the construction of sensorial structures is depicted in Fig. 1. First, a geometric CAD model of the desired technical structure is created. Based on the model, a library of standard load cases which are expected during operation as well as exceptional overloading or misuse cases is defined. This library can in principle comprise several hundred load cases, e.g., various forces and temperatures applied at various positions. FEM simulations then are run on the model, generating a large database of effect data (e.g., displacements, strains, stresses, temperature values etc.) which, depending on the complexity of the FEM model and the number of defined load cases, can be several gigabytes in size and may take several days to compute. It is, of course, impractical to directly run load case inference algorithms on such a large amount of pre-calculated data during actual operation of the structure, let alone to embed it into the structure using ROM circuits. Therefore, data compaction and data mining algorithms [5], [6] need to be applied to the FEM input and output in an offline preprocessing step to yield a considerably reduced dataset of key figures much fewer in count but of high informative value in terms of reliable load case estimation and overload detection. The simplest solution here would be to let a human expert choose selected sensor positions based on the geometric characteristics of the structure and manually rig the model with a desired distribution of sensors for assessing strain, pressure, temperature or similar effects that are present in the FEM calculation output. Mathematical optimization methods [7], [8] can be used to aid finding optimal sensor distributions at this stage. If a good sensor distribution has been found, a physical prototype equipped with real sensors is built, to which (a subset of) the previously defined load case library is physically applied in a series of training experiments. The sensor signals resulting from these load cases are recorded and stored along with the respective FEM results in the effects database. The same data reduction procedures that are performed on the FEM output variables can be executed on the measured sensor values. For example, key figure calculation rules producing statistics (e.g., minimum, maximum, average value, standard deviation, or much more complex computations) over selected groups or multilevel hierarchies of sensors could be defined or automatically learned using machine learning and data mining techniques. It is an open research topic how such calculation rules can be automatically generated from the effects and sensor value database in the most beneficial way with respect to a given load case inference problem.

The resulting compact dataset is fed as training data into suitable machine learning algorithms [6], [9], [10]. The focus in this step lies on algorithms whose learned models as well as respective model evaluation procedures are simple and small enough to be directly transformed into miniaturized low-power hardware circuits. Embedding these circuits into the material yields a sensorial structure which is able to constantly monitor its loading state during operation and issue alerts to its user when overloading is detected.

III. Functional Mockup System

To practically evaluate the described process, a functional mockup system was developed which allows to experiment with different sensor assessment, data mining, and machine learning methods. As a first evaluation scenario, we use a simple St37 steel plate with the dimensions $200 \times 300 \times 1$ mm mounted with a fixture along one short side and a movable single point support at the bottom face. Different weights can be placed on the top face. The objective is to estimate the locations and masses of the weights and recognize overload situations only by examining a few strain values

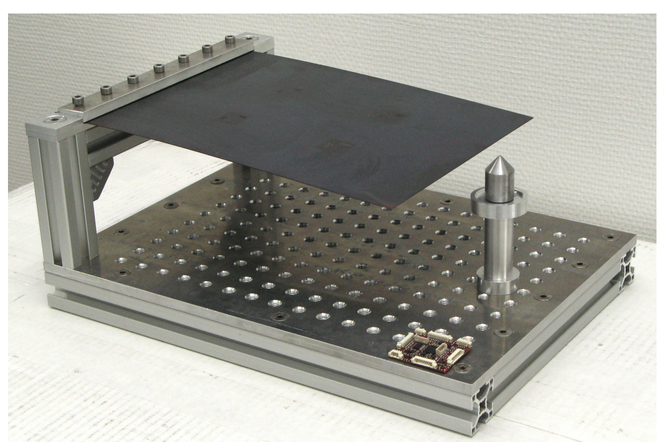

Figure 2. Metal plate setup

ISBN: 978-1-61208-154-0
DOI: 10.13140/2.1.3124.0647

measured at the plate's bottom face using machine learning algorithms. For this purpose, four pairs of orthogonally aligned strain gauges are printed at selected positions onto a polymer insulation layer applied to the bottom face. Each pair measures the strain in the X and Y direction of the local face coordinate system and is connected to a miniaturized sensor node which provides analog signal conditioning as well as digital data processing and message based communication implemented in FPGAs. The physical demonstrator built for this scenario permits the installation of plates with different dimensions, material properties, and sensor setups. Each possible sensor configuration consists of sensor nodes arranged in a topologically two-dimensional mesh network with each node having direct connections to up to four neighbors. For example, Fig. 2 shows the demonstrator equipped with a $200 \times 245 \times 1$ mm steel plate and four pairs of Ag strain gauges printed on the plate's top face in an Aerosol Jet® printing process [11] using commercial pure silver ink. With material combinations permitting the increase of sintering temperatures, a future switch to CuNi alloys with less temperature-dependent resistivity is envisaged. In the present case, insulation between the printed conductive paths and the steel substrate was provided by a silicone-based coating layer, once more applied by means of Aerosol Jet® printing.

The load case library of the experimental setup is defined by dividing the top face of the 200×300 mm plate into a 20×20 mm grid of 176 equidistant load positions. At each position weights of 100 g and 200 g are individually applied, resulting in 352 different load cases with a single weight each. As the sensor inputs directly depend on the strains occurring in the material, overloading is defined in terms of displacements instead of strains. This adds a further complication to the evaluation scenario: We aim at investigating how good the tested machine learning algorithms perform at expressing correlations between different types of mechanical effects. Thus, overloading is defined to occur if and only if at least one point on the plate's bottom face moves downward in global Z direction more than a given threshold t_z. To prevent a bias towards positive or negative classifications in the training set, the load case library is partitioned into 50 % overload and 50 % non-overload situations by obtaining the maximum Z displacement for each load case from the FEM output and setting the threshold to the median of these 352 values, yielding $t_z = 130.6\ \mu m$ for this particular scenario. Here, the desired result from the machine learning stage is a—preferably simple—mapping from the measured sensor values to estimated X and Y coordinates of the applied load (in the local surface coordinate system), an estimated mass of the weight in grams, and a yes/no classification whether the displacement threshold t_z has been exceeded by any point of the plate. Accordingly, the desired physical result is an actual low-power microchip that implements this mapping in hardware.

IV. A Robust Sensor Network

When building structures that monitor their own health state by means of spatially distributed sensors, it is important that the introduction of local defects in the communication links between evaluation nodes, e.g., caused by overloading or misuse, does not result in entire regions of the structure being cut off from communication and, consequently, global monitoring processes. Hence, the communication infrastructure for sending sensor values and issued alerts across the structure must be robust with respect to sudden failure of communication links. For this purpose, we have implemented a sensor network capable of dynamically routing data packets across irregular sensor node topologies, e.g., caused by damage of communication links. Such an irregular network topology is shown in Fig. 3. Network nodes are numbered from 0 to 12; communication links are depicted as arrows.

Each sensor node in the network digitizes the signals of a small local group of physical sensors (e.g., two to eight in count) and is connected to spatially neighboring nodes via point-to-point serial links that define the network topology. To enable the usage of unmatched and uncalibrated sets of sensors, the signal processing is implemented with a zooming window analog-to-digital converter (ADC) approach. Spatial proximity relations among the physical sensors are directly translated to proximity relations in the network topology. With respect to communication, all nodes simultaneously act as routers and communication endpoints, with unique topological coordinates assigned to each of them. Communication is message based using the Simple Local Intranet Protocol (SLIP) [12] which employs dynamic routing strategies to forward data packets from their source nodes to desired destination nodes in the network. For this purpose, each packet contains a discrete multi-dimensional vector specifying the packet's destination in terms of remaining distance in the network topology. All routing information is held in the packets without any additional routing tables maintained by the sensor nodes. The SLIP communication protocol stack is implemented in hardware as a System-on-a-Chip architecture using FPGA/ASIC target technologies,

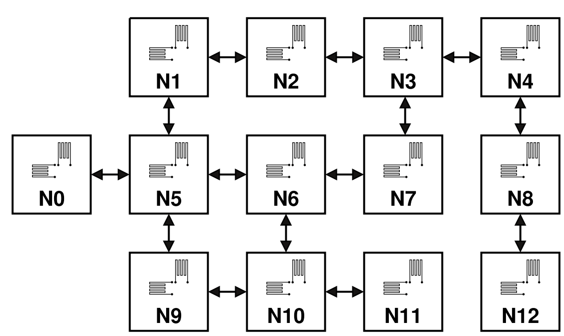

Figure 3. Irregular sensor network topology

ISBN: 978-1-61208-154-0
DOI: 10.13140/2.1.3124.0647

and in software using the C and ML programming languages as well as the ConPro behavioral parallel programming language in conjunction with the ConPro synthesis framework [13]. The communication protocol is freely scalable in the network's size and topological dimensionality.

V. k-Nearest-Neighbor and Decision Tree Learning

Adhering to the design constraints imposed by the application scenario, experiments were started with an implementation of two of the most basic classification and regression algorithms. Concretely, C4.5 decision tree learning [9] and the simple k-Nearest-Neighbor algorithm [10] were examined.

The training set contains all correct load positions, masses and classification results for every load case in the load case library along with the respective 8-dimensional strain vector determined via either sensor measurement or FEM simulation. To answer a query with k-Nearest-Neighbor (k-NN), the k entries closest to the query strain vector are selected and combined into a single estimation for the target variables. Numerical regression of the load position and mass can be achieved by calculating the weighted average, whereas discrete (e.g., Boolean) classification is possible by means of weighted voting. For a fixed k, let $E_1, \ldots, E_k$ denote the k nearest points in the training set, (x_i, y_i, m_i) the load coordinates and mass of E_i, and $Q \in \mathbb{R}^8$ the query vector. The estimated load position and mass (x_Q, y_Q, m_Q) for Q is calculated using inverse distance weighting, i.e.,

$$\begin{pmatrix} x_Q \\ y_Q \\ m_Q \end{pmatrix} = \sum_{i=1}^{k} w(E_i) \cdot \begin{pmatrix} x_i \\ y_i \\ m_i \end{pmatrix} \quad (1)$$

$$\text{with } w(E_i) = 1/(\mathrm{dist}^2(Q, E_i) \cdot w_{\mathrm{sum}}),$$
$$w_{\mathrm{sum}} = \textstyle\sum_{i=1}^{k} 1/(\mathrm{dist}^2(Q, E_i))$$

and $\mathrm{dist}^2(\mathrm{Q}, \mathrm{E_i})$ being the squared Euclidean distance between points Q and E_i in 8-dimensional strain space.

Weighted voting is done by adding the weighting factors $w(E_i)$ of all E_i that have the same classification and then assigning to Q the classification which gained the largest sum, defaulting to a "safe" answer (yes) in case of a tie.

In our first experiments, the C4.5 decision tree learning is employed for classification only as its reliable extension to multivariate numerical regression is more elaborate.

VI. Evaluation Results

Initial tests with the steel plate displayed in Fig. 2 and loads limited to the material's elastic range showed that the signal quality attained with printed Ag gauges and a first version of the sensor node ADCs was not yet suitable for input into machine learning processes. As an intermediate step, trials with a nitrile rubber plate and glued-on commercial constantan strain gauges were made. In that scenario, higher strains and, thus, increased signal ranges lead to relatively usable load identification results [14]. However, the time-dependent hyperelastic behavior of polymers is highly problematic with respect to long-term usage: The prediction accuracy of learned models as well as the consistency between FEM output and the plate's actual physical state deteriorates after a certain number of load cycles. Hence, structures made from such materials require machine learning models that, depending on the individual loading history, automatically adjust their output to the structure's aging process. The design and synthesis of machine learning processes capable of generating such self-adapting models from FEM output, experimental data, and explicit mechanical domain knowledge poses a challenging task.

With the revision of the physical metal plate setup still pending, the evaluation presented in this paper uses completely noise-free strain values from the FEM simulation step as input to the machine learning process to gain a first set of reference success rates for the 200×300 mm steel plate scenario and the two described machine learning algorithms. In this case, the learned models stay valid as long as all load-induced stresses fall within the elastic range. As the prediction accuracy can be regarded as a measure of the general practicability of our approach, the success rates obtained in this evaluation will be used as benchmarks for further improvement of the hardware prototype as well as development of more sophisticated machine learning approaches.

Since only values from FEM simulation are used in the evaluation, i.e., no comparison with measured values from the physical prototype is made, the spike placed under the plate's bottom face was modeled as a roller support in the FEM model for sake of simplicity. For C4.5 and k-NN with $1 < k < 10$, a leave-one-out cross-validation (LOOCV) was conducted on the training set described in Section III: Each singleton subset of the training set was used exactly once for querying the machine learning models constructed from its complementary set and comparing the result with the known correct values. In addition, the models obtained from the entire 352 element training set were queried using an intermediate test set INT consisting of 150 g weights placed exactly at the midpoints between the previously defined grid positions, resulting in further 10×15 load cases neither the coordinates nor masses of which appear anywhere in the training set. Depending on the value chosen for k, the percentage of correct classifications achieved with k-NN varies between 89.77 % and 90.91 % in LOOCV and between 92 % and 93.33 % for INT. The C4.5 decision tree attained 94.32 % correct classifications in LOOCV and 96.22 % for INT. The locations at which false positives and false negatives were returned in LOOCV by C4.5 and k-NN with $k = 4$ are shown in Fig. 4. While at each position a 100 g and a 200 g load case were tested, no position

ISBN: 978-1-61208-154-0
DOI: 10.13140/2.1.3124.0647

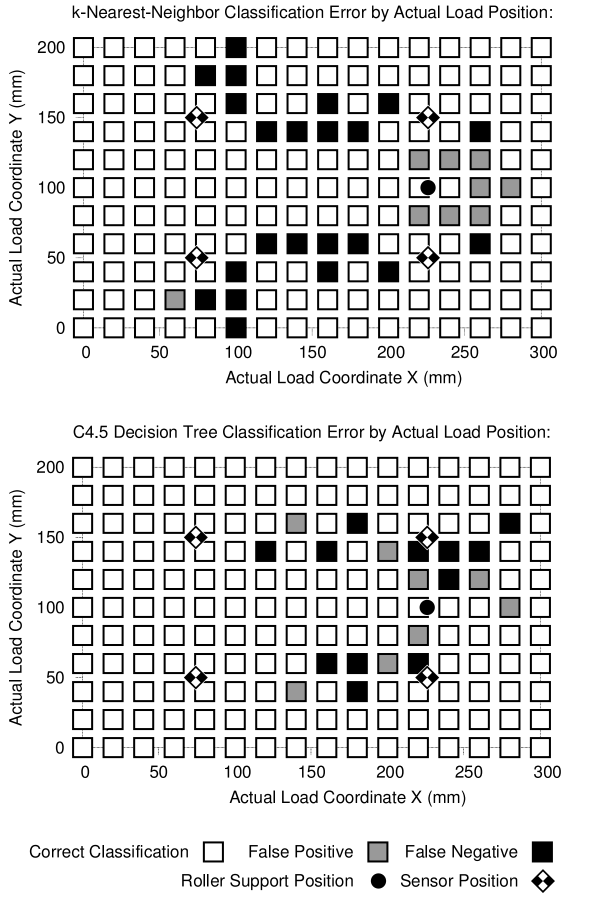

Figure 4. Classification results obtained with leave-one-out cross-validation on the training set

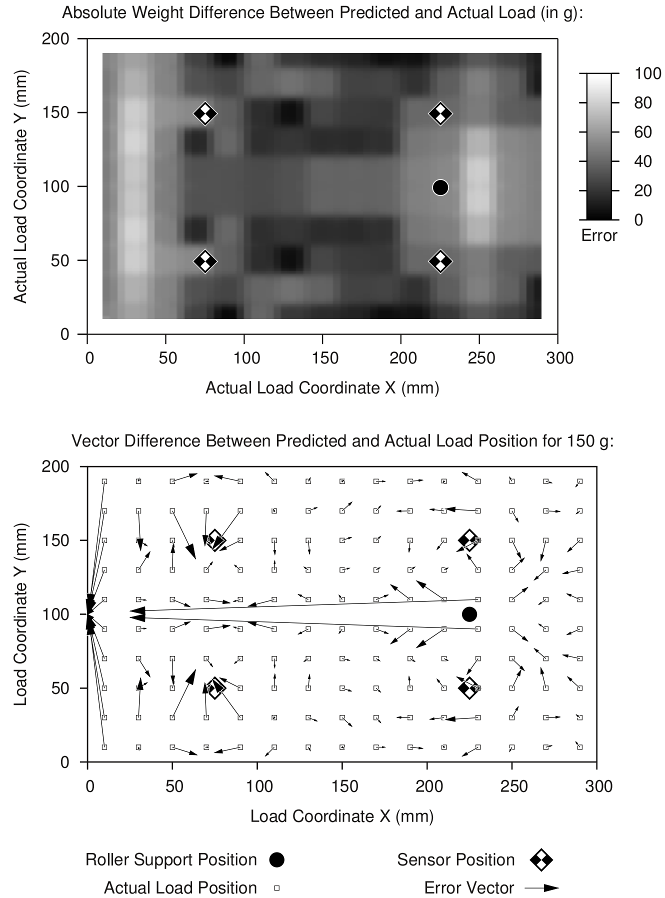

Figure 5. Regression error for 150 g weights placed at intermediate positions

produced wrong classifications for both masses.

The numerical regression results obtained for the test set INT by k-NN with $k = 4$ are illustrated in Fig. 5. The difference vectors from the actual to the estimated load positions are shown in the bottom vector field (median length 9.09 mm, average length 15.88 mm), whereas the absolute error in load mass estimation is visualized in the upper plot (median 20.39 g, average 25.38 g). Among all tested values for k, the median difference vector length of the load coordinate regression was smallest with $k = 4$, while the average was smallest with $k = 2$. In Fig. 4 and 5, the locations of the strain sensors and the roller support are indicated with diamond and circle symbols, respectively. The fixture is located on the Y coordinate axis. It is clearly noticeable that the regression error is largest in the vicinity of the supports.

VII. Monitoring More Complex Structures with Intelligent Agents

While the evaluation results suggest that the tested machine learning methods are, in principle, suitable for load case estimation on simple structures like a metal plate, it is not clear how these methods can successfully be utilized for monitoring much more complex technical structures as well, e.g., objects that comprise many, possibly movable, parts. Several problems arise in such application scenarios. Firstly, the definition of a comprehensive and adequate load case library is more difficult for these structures as the number of relevant load cases might increase exponentially with the number of constituent substructures and the mechanical effects might depend on the movable parts' positions and orientations. Secondly, even if the load case library or FEM effects database does not already reach a prohibitive size due to combinatorial growth, the more complicated the cause-effect relations reflected in the produced datasets become, the less fruitful an identification and extraction of these patterns into a proper model may prove in the data mining and machine learning stages. As a consequence, higher error rates are to be expected in the actual load case inference. Finally, the machine learning model for the entire structure may grow to a size where it neither can be created or queried as a whole in reasonable time with the available resources nor fit on a single chip. To circumvent these difficulties, we propose an Intelligent Agent based approach to solving the global monitoring problem by splitting it into multiple spatially constrained subproblems and solving these in a distributed fashion.

Generally, an agent can be viewed as a hardware and/or software entity that receives perceptions from an environment it is situated in and reacts to them, either reflex-based [15] or deliberately [16], in order to perform one or more specific tasks [17]. Applied to the task of monitoring a sensorial structure, the agent may be physically situated on the latter, with its perceptions coming from different groups of sensor nodes and communication with other agents. If the agent has access to a knowledge base of facts about the structure's geometric and mechanical characteristics as well as cause-effect relations with respect to load introduction, it can draw conclusions about the current state of its environment based on its sensory input, i.e., make sense of the sensor signals. It is the presence of this semantic level that distinguishes intelligent agents from nodes of the sensor network described in Section IV: While the latter only provide the infrastructure for routing measured sensor data across the physical object and have no way of interpreting this data, the agents are capable of incrementally constructing a mental model of their surroundings over time.

In our approach, the structure is partitioned into connected substructures, which can be further partitioned into regions. To each region an individual monitor agent is assigned that regularly adjusts its internal model of the respective region according to its perceptions. Communication among agents covering directly neighboring regions enables the construction of a (simplified) distributed global view of the entire structure. For this purpose, each monitor agent requires an appropriate description of how loads and their effects are transmitted across the boundaries between the agent's region and each of the adjacent regions. That way, an agent can infer from its local sensory input the structural state at these boundaries and send the results to its topological neighbors, which in turn update their local models by combining their own sensor data with the boundary state information they received from their respective neighbors, and, again, pass the updated boundary states on to these, and so forth. Establishing a multilevel hierarchy of agents and incorporating (to a manageable extent, given the available resources) equation-based knowledge from the FEM domain into the reasoning process may yield a distributed load inference algorithm that is adaptive in terms of desired temporal resolution and predictive precision, depending on the number of communication rounds performed per time unit. Thus, one of the next logical steps in our research will be the development of such algorithms along with the necessary agent communication protocols and their evaluation in the functional mockup system.

VIII. Conclusion and Future Work

This article presented a novel AI-based process for the construction of self-monitoring sensorial structures that utilizes machine learning methods for resource-constrained real-time load case inference. A functional mockup system using a simple steel plate as evaluation scenario was introduced and the general practicability of the machine learning approach was shown using noise-free data from FEM simulation only. In the evaluation very simple algorithms like k-NN and decision tree learning already yielded over 90 % correct classifications in the detection whether the plate's maximum displacement vector exceeded a given length due to load introduction. Also numerical regression of the load locations and masses with k-NN attained a prediction quality that may be acceptable in many practical application cases. It seems likely that the results can be improved by employment of more advanced machine learning methods and, in particular, by incorporation of specific domain knowledge from the field of applied mechanics into these. The focus here should lie on the best possible elimination of false negatives, which correspond to unrecognized overload situations, and on the increase of prediction quality for sensor signals with a low signal-to-noise ratio. On the hardware side, further improvement of functional printing processes as well as miniature ADC components is expected to lead to better signal quality and, thus, more accurate load identification results. In addition to this, one of the next steps in our research will be the development and implementation of multi-agent based approaches that enable monitoring of geometrically much more complex structures. When the viability of this has been shown, the machine learning models need to be enhanced to automatically accomodate to structural aging while in operation, e.g., in materials like polymers or textiles, where this aspect is of particular importance.

Acknowledgment

The authors gratefully acknowledge the financial support by the State and University of Bremen in the framework of the ISIS (Integrated Solutions in Sensorial Structure Engineering) Sensorial Materials Scientific Centre (fund numbers 54416906, 54416918, and 54416919).

References

[1] M. Maiwald, C. Werner, V. Zöllmer, and M. Busse, "INKtelligent Printing® for sensorial applications," *Sensor Review*, vol. 30, no. 1, pp. 19–23, 2010.

[2] D. Mayer, T. Melz, C. Pille, and F.-J. Wöstmann, "CASTTRONICS – direct integration of piezo ceramic materials in high pressure die casting parts for vibration control," in *Proceedings of Actuator 2008 – 11th International Conference on New Actuators*, Bremen, Germany, June 9th-11th 2008, pp. 61–66.

ISBN: 978-1-61208-154-0
DOI: 10.13140/2.1.3124.0647

[3] V. K. Wadhawan, *Smart Structures: Blurring the Distinction between the Living and the Nonliving*. Oxford Univ. Press, 2007.

[4] R. J. M. Vullers, R. van Schaijk, I. Doms, C. Van Hoof, and R. Mertens, “Micropower energy harvesting,” *Solid State Electron.*, vol. 53, no. 7, pp. 684–693, 2009.

[5] O. Maimon and L. Rokach, Eds., *The Data Mining and Knowledge Discovery Handbook*. Springer, 2005.

[6] A. J. Izenman, *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. Springer, 2008.

[7] R. Jedermann and W. Lang, “The minimum number of sensors - interpolation of spatial temperature profiles,” in *Proceedings of the 6th European Conference on Wireless Sensor Networks (ESWN)*, U. Roedig and C. Sreenan, Eds. Springer, 2009, pp. 232–246.

[8] A. R. M. Rao and G. Anandakumar, “Optimal placement of sensors for structural system identification and health monitoring using a hybrid swarm intelligence technique,” *Smart. Mater. Struct.*, vol. 16, no. 6, pp. 2658–2672, 2007.

[9] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufman Publishers, Inc., 1993.

[10] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[11] M. Maiwald, C. Werner, V. Zöllmer, and M. Busse, “INKtelligent® printed strain gauges,” in *Proceedings of the Eurosensors XXIII conference*, Lausanne, Switzerland, September 6th-9th 2009, pp. 907–910.

[12] S. Bosse and D. Lehmhus, “Smart communication in a wired sensor- and actuator network of a modular robot actuator system using a hop-protocol with delta routing,” in *Proceedings of Smart Systems Integration 2010 - 4th European Conference and Exhibition on Integration Issues of Miniaturized Systems*, T. Gessner, Ed., Como, Italy, March 23st-24rd 2010, CD-ROM.

[13] S. Bosse, “Hardware-software-co-design of parallel and distributed systems using a unique behavioural programming and multi-process model with high-level synthesis,” in *Proceedings of SPIE Microtechnologies Conference 2011, Session EMT 102 VLSI Circuits and Systems*, Prague, Czech Republic, April 18th-21th 2011.

[14] F. Pantke, S. Bosse, D. Lehmhus, M. Lawo, and M. Busse, “Combining simulation and machine-learning for real-time load identification in sensorial materials,” in *SimBio-M 2011 – Simulations in Bio-Sciences and Multiphysics International Conference*, Marseille, France, June 20th–22nd 2011, accepted.

[15] R. A. Brooks, “Elephants dont play chess,” *Robot. Auton. Syst.*, vol. 6, no. 1&2, pp. 3–15, 1990.

[16] A. S. Rao and M. P. Georgeff, “Decision procedures for BDI logics,” *J. Logic Comput.*, vol. 8, no. 3, pp. 293–343, 1998.

[17] N. R. Jennings, “On agent-based software engineering,” *Artif. Intell.*, vol. 117, no. 2, pp. 277–296, 2000.