Compressed Sensing and Time-Parallel Reduced-Order Modeling for Structural Health Monitoring Using a DDDAS

J. Cortial[1], C. Farhat[1,2], L.J. Guibas[3], and M. Rajashekhar[3]

[1] Institute for Computational and Mathematical Engineering
[2] Department of Mechanical Engineering
[3] Department of Computer Science
Stanford University
Stanford, CA 94305, U.S.A
jcortial@stanford.edu,cfarhat@stanford.edu,
guibas@cs.stanford.edu,manj@stanford.edu

Abstract. This paper discusses recent progress achieved in two areas related to the development of a Dynamic Data Driven Applications System (DDDAS) for structural and material health monitoring and critical event prediction. The first area concerns the development and demonstration of a sensor data compression algorithm and its application to the detection of structural damage. The second area concerns the prediction in near real-time of the transient dynamics of a structural system using a nonlinear reduced-order model and a time-parallel ODE (Ordinary Differential Equation) solver.

1 Introduction

The overall and long-term goal of our effort is to enable and promote active health monitoring, failure prediction, aging assessment, informed crisis management, and decision support for complex and degrading structural engineering systems based on dynamic-data-driven concepts. Our approach involves the development a Dynamic Data Driven Applications System (DDDAS) that demonstrates numerically, as much as possible, its suitability for structural health monitoring and critical event prediction. An outline of this approach, its objectives, and preliminary architectural concepts to support it can be found in [1].

The main objective of this paper is to describe progress achieved in two areas of activity that are directly related to drastically reducing the overall sensing and computational cost involved in the utilization of such a DDDAS. For this purpose, our efforts draw experiences and technologies from our previous research on the design of a data-driven environment for multiphysics applications (DDEMA) [2,3].

The first area of progress pertains to the development and implementation of an efficient data compression scheme for sensor networks that addresses the issue of limited communication bandwidth. Unlike other data compression algorithms,

Y. Shi et al. (Eds.): ICCS 2007, Part I, LNCS 4487, pp. 1171–1179, 2007.
© Springer-Verlag Berlin Heidelberg 2007

this scheme does not require the knowledge of the compressing transform of the input signal. It constitutes a first step towards the efficient coupling between a given sensor network and a given computational model.

The second area of progress pertains to the development and implementation of a numerical simulator for the prediction in near real-time of the transient dynamics of a structural system. The new aspect of this effort is the generalization to nonlinear second-order hyperbolic problems of the PITA (Parallel Implicit Time-integration Algorithms) framework developed in [4,5] for linear time-dependent problems.

Preliminary verification and performance assessment examples that furthermore illustrate the basic concepts behind both methodologies outlined above are also provided in this paper.

2 An Efficient Data Compression Scheme for Sensor Networks

In the context of a DDDAS for health monitoring and critical event prediction, the sensor network collects the monitored data of the structure and sends it to the numerical simulator. This simulator is equipped with full- and reduced-order computational models that are validated for healthy states of the structure. Differences between sensed and computed (possibly in near real-time) data, or sensed data and data retrieved in real-time from a computational data base in the simulator, form a special class of sparse signals that can be used to locate, assess, and possibly predict the evolution of structural damage as well as dynamically update the computational models [6,7]. Hence, the sensor network acts in general as the source of the data used for guiding and driving numerical simulations and/or updating the underlying computational models.

Communication bandwidth is a very limited resource in sensor networks. For this reason, the trivial approach where all sensor readings are sent to the simulator is unlikely to be deployed. Compressed sensing offers a venue to a different approach that avoids expensive communication between the sensor network and the numerical simulator. Such an alternative approach is not only more practical for health monitoring in general, but also essential for critical event prediction. In this section, we discuss our chosen data compression scheme whose theory was developed in [8]. Unlike many other data compression algorithms, this scheme does not require any information about the compressing transform of the input signal. It only assumes that the signal is compressible — that is, it has a sparse representation in some orthogonal basis. In health monitoring applications, this assumption typically holds given that as mentioned above, a signal usually consists of the difference between a measured information and a predicted one. Since a sparse signal can be reconstructed from the knowledge of a few random linear projections [8], collecting only these few random projections constitutes a feasible mechanism for data compression and provides the sought-after saving in communication time.

2.1 Compressed Sensing Overview

Let $x_0 \in \mathbb{R}^m$ denote a signal of interest that has a sparse representation in some orthogonal basis, and $\phi \in \mathbb{R}^{n \times m}$, with $n < m$, denote the random matrix generated by the "uniform spherical ensemble" technique described in [8]. Consider the vector $y = \phi x_0$. This matrix-vector product represents the result of n random linear projections — or samples — of the original signal x_0. From the knowledge of y, one can reconstruct an approximation $\tilde{x}_0$ of x_0 as follows

$$\tilde{x}_0 = \arg\min_x ||x||_1 \text{ subject to } y = \phi x \tag{1}$$

The reconstruction technique summarized by the above equation involves essentially linear, non-adaptive measurements followed by a nonlinear approximate reconstruction. In [8], it is proved that despite its undersampling ($n < m$), this technique leads to an accurate reconstruction $\tilde{x}_0$ when $n = O(N \log(m))$, where N denotes the number of largest transform coefficients of x_0 — that is, the number of transform coefficients of x_0 that would be needed to build a reasonable approximation of this signal. For example, for a k-sparse system — that is, a system with only k values different from zero or larger in absolute value than a certain minimum threshold — $N = k$.

2.2 Implementation

In our proposed sensing approach, we avoid expensive communication between the sensor network and the simulator by communicating to the latter only summaries of the original sensor readings. These summaries contain however enough information to allow the extraction of the deviant sensor readings and localize the sensors announcing deviant information. Here, "deviant" refers to a measurement that is significantly different from an expected value obtained by simulation and therefore is indicative, for example, of structural damage.

More specifically, our approach to compressed sensing is composed of two main steps that are described next.

Extraction of Compressed Summaries. Let $s_0 \in \mathbb{R}^m$ and $p_0 \in \mathbb{R}^m$ denote the measured and numerically predicted readings at m sensor nodes, respectively. We compute n ($n << m$) random linear projections of these readings as follows

$$y_s = \phi s_0 \,,\ y_p = \phi p_0 \tag{2}$$

The n compressed measurements of sensor readings represented by y_s are then communicated over the network to the simulator. Since $n << m$, this leads to a reduction in communication time.

Decoding of a Compressed Representation. At the simulator, we compute the difference between the sensor readings and corresponding simulator predictions

$$\delta y = y_s - y_p = \phi s_0 - \phi p_0 = \phi(s_0 - p_0) \tag{3}$$

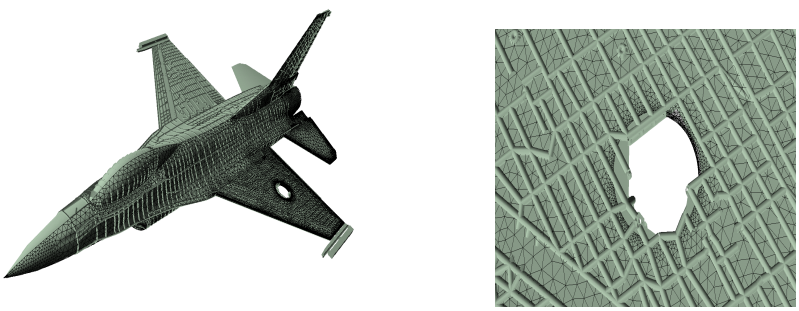
For a computational model that is validated for a healthy state of the structure, the simulator predictions should be in principle in reasonable agreement with the sensor readings most of the time. Therefore, the signal $(s_0 - p_0)$ is in principle a sparse and therefore compressible one that can be accurately reconstructed by solving the minimization problem underlying Eq. (1) after substituting x by $(s - p)$ and y by δy from Eq. (3). Then, as demonstrated in Section 2.4, the reconstructed value $(\tilde{s}_0 - \tilde{p}_0)$ of $(s_0 - p_0)$ can be exploited, after noise removal, to localize the deviant sensor readings, predict the location of damage and possibly characterize it.

2.3 Summary

The compressed summaries outlined above behave like an *error detecting code*. They allow the system to detect events of interest, which facilitates automatic detection. They also contain enough information to allow the localization of the sensor nodes contributing to deviant readings. Finally, compressed sensing allows faster data acquisition by lowering the communication burden.

2.4 Preliminary Verification

To illustrate the compressed sensing methodology described above, we consider here the case of an F-16 fighter jet that is suddenly hit by an explosive bullet in one of its wings. For simplicity, we ignore the damage formation process in order to focus on the compressed sensing issues. Hence, we simulate here the inflicted damage by suddenly introducing in the Computational Structural Dynamics (CSD) and Computational Fluid Dynamics (CFD) models a realistic hole that alters the stiffness, mass, and aerodynamic properties of the aircraft (see Fig. 1). We equip the simulator with validated, full- and reduced-order, aeroelastic computational models of the F-16 such as those presented in [9,10,11]. We assume that 9131 sensor nodes are deployed over the entire fighter jet. We note



(a) CFD (wireframe) & CSM (shaded) (b) Structural damage

Fig. 1. CFD and CSM models of an F-16 fighter with a hole in the port side wing

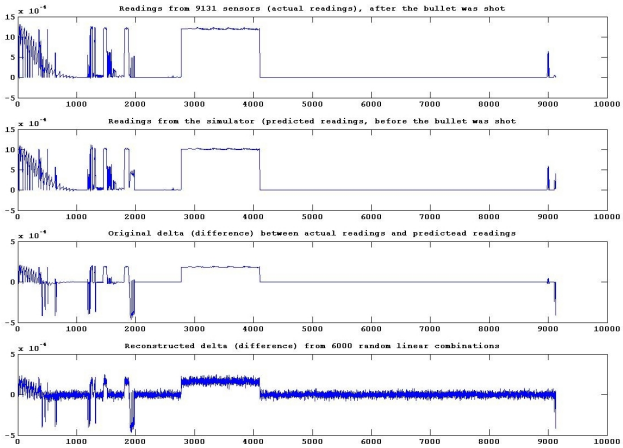


Fig. 2. In all plots, the x-axis represents the sensor nodes and the y-axis represents the sensed, predicted, difference, and reconstructed difference values of the vertical component of the displacement field at the sensor nodes

that this massive instrumentation hypothesis is currently unrealistic because of sensor weight issues. We are currently addressing this aspect of the problem by conducting various trade-off studies. We simulate the data measured by the sensor nodes after damage infliction using the AERO simulator [9,10] and compare it to the pre-damage data obtained by numerical predictions using the validated computational models.

Fig. 2 reports our findings using the compressed sensing approach with $m = 6000$ — that is, when collecting 6000 instead of 9131 readings. The peak values of the reconstructed difference signal shown in the bottom most diagram of this figure correlate well with the locations of the sensor nodes that are closest to the region hit by the explosive bullet. This demonstrates the potential of compressed sensing for locating structural damage. Using such a massive instrumentation also allows the in-flight estimation of the size of the damage, among other properties.

3 A Time-Parallel Computational Framework for Faster Numerical Predictions

The transient dynamics of a complex structural system are typically modeled by a set of Partial Differential Equations (PDEs) that are semi-discretized by the finite element method. The solution of the resulting semi-discrete systems of equations is typically organized around two loops: (a) an inner-loop which evaluates some quantities related to the state of the structure at a specific time-instance, and (b) an outer one which advances the state of the structure from a given time-instance to the next one. Hence, the computational cost associated

with the time-integration of the semi-discrete systems governing structural dynamics problems can be divided into two main components: (a) the CPU time incurred by the inner-loop — that is, by the evaluation of the displacement field and/or related quantities at a given time-instance, and (b) that incurred by the outer-loop — that is, by time-advancing the complete state of the structure. To a large extent, parallel algorithms for the solution of transient dynamics problems have mainly addressed the reduction of the CPU time associated with the inner-loop. Such parallel algorithms can be described as space-parallel algorithms. By comparison, the parallelization of the outer-loop of a structural dynamics time-integrator — or time-parallelism — has received little attention in the literature. This is because, in general, time-parallelism where the solution is computed simultaneously at different time-instances is harder to achieve than space-parallelism, due to the inherently sequential nature of the time-integration process. However, time-parallelism can be of paramount importance to fast computations, for example, when space-parallelism is unfeasible, cannot exploit all the processors available on a given massively parallel system, or when predicting structural dynamics responses in near-real-time requires exploiting both space- and time-parallelisms.

Usually, space-parallelism is unfeasible when the number of degrees of freedom (dofs) is too small to allow amortization of the interprocessor communication overhead. Reduced-order computational models, which can be a critical component of an on-board DDDAS, examplify this scenario as they often involve only a few dofs. Yet, these and other computational models with relatively few dofs can be (or feel like they are) CPU intensive when time-integration is performed over a relatively large time-interval, and/or real-time predictions are desired, for example, for on-line critical event prediction or control purposes. Predicting in near-real-time the time-evolution of the variables governed by such models calls for time-parallelism. Finally, when considering off-line transient structural dynamics computations on massively parallel computers with thousands of processors such as the Department of Energy's ASCI machines, space-parallelism alone is not always the optimal strategy for reducing the time-to-solution [4]. Indeed, in many cases such as low-frequency applications, the mesh resolution dictated by accuracy considerations can be accommodated by a fraction of the total number of available processors. It is well-known that in such cases, because of hardware-related issues such as memory bandwidth effects and interprocessor communication costs, the solution time typically increases when the number of processors N_p is increased beyond a certain critical value, N_p^{cr}, while the mesh size is maintained constant. In this sense, the smaller is N_p^{cr} compared to the maximum number of available processors, N_p^{max}, the less optimal is the space-parallel implementation strategy, and the more adequate becomes an approach that combines both space- and time-parallelisms in order to reduce as much as possible the solution time by exploiting all N_p^{max} processors [4].

During the last decade, at least three approaches have been proposed for time-parallelizing time-integrators (for example, see [4] for a brief review). Among these approaches, the second-generation PITA [5] stands out as the only

computational approach that has demonstrated a potential for accelerating the solution of second-order hyperbolic problems such as those encountered in structural dynamics applications. However, this computational framework was developed in [5] for linear problems only. Under this DDDAS research effort, we have recently generalized it to nonlinear dynamics problems such as those encountered in failure prediction and aging assessment.

3.1 Nonlinear Structural Dynamics Computational Models

Nonlinear structural dynamics computational models can often be written as

$$M\ddot{u} + f^{int}(\dot{u}, u) = f^{ext}(t)\,,\; u(t_0) = u_0\,,\; \dot{u}(t_0) = \dot{u}_0 \tag{4}$$

where M, u, and $\dot{u}$ denote the mass matrix, and displacement and velocity vectors, respectively, f^{int} denotes the vector of internal forces which typically vary nonlinearly with u and $\dot{u}$ and f^{ext} denotes the vector of external forces. In the context of a DDDAS, Eq. (4) represents a full-order nonlinear computational model when discussing off-line simulations and a reduced-order model (ROM) when discussing on-line simulations.

3.2 Nonlinear PITA

The nonlinear PITA framework for the time-parallel solution of Eq. (4) is an iterative one. It can be described as follows. The time-domain of interest is partitioned into time-slices whose boundary points are treated as a coarse time-grid. In a pre-processing step, the solution is approximated on this coarse time-grid to provide a seed — that is, an initial condition — for each time-slice, using a sequential but computationally inexpensive numerical procedure. Next, the following computational steps are performed. First, a preferred time-integrator is applied independently and therefore concurrently in each time-slice to advance the solution from the starting point of this time-slice to its end point. This introduces jumps in the computed solution at the interior points of the coarse time-grid. Then, a Newton-based corrective step is performed on the original time-grid to improve the accuracy of the seeds and reduce the magnitude of these jumps. This two-step sequence of computations is repeated until convergence is reached — that is, until the solution jumps at the interior points of the coarse time-grid are eliminated. The computational burden associated with the corrective Newton step is alleviated by the introduction of a carefully constructed projector, so that the sequential computational complexity of each iteration of a nonlinear PITA is only slightly larger to that of its underlying (or preferred), time-sequential, time-integration algorithm. Therefore, the asymptotic parallel speedup of the nonlinear PITA using N_p processors is roughly equal to N_p/N_{it} where N_{it} denotes the number of iterations for convergence. This result reveals that the nonlinear PITA is mostly interesting when N_{it} can be kept as small as possible, say of the order of 2 to 4.

3.3 Preliminary Performance Assessment

To illustrate the potential of the recently developed nonlinear PITA for near real-time simulations such as those envisioned on an on-line DDDAS, we consider here the solution of Eq. (4) in the context of a reduced-order model of the F-16 aircraft shown in Fig. 1 and a cyclic (fatigue) load. We choose to equip PITA with the midpoint rule as the underlying time-integrator and decompose the time-domain of interest in 20 time-slices. The computational results reported in Fig. 3 reveal that for the considered problem, the nonlinear PITA converges in 3 iterations, which highlights its potential for supporting an on-line DDDAS for health monitoring.

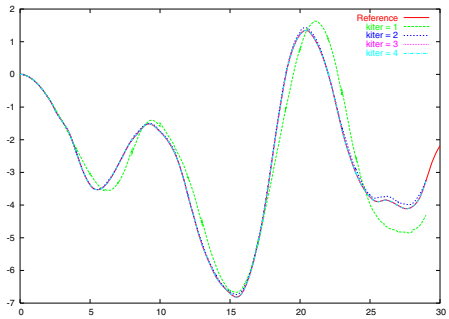


Fig. 3. Convergence of the nonlinear PITA equipped with the midpoint rule and applied to the solution of Eq. (4) in the context of an on-line reduced-order model

Acknowledgement. The authors acknowledge the support by the National Science Foundation under Grant CNS-0540419.

References

1. Farhat, C., Michopoulos, J. G., Chang, F. K., Guibas, L. J., Lew, A. J., Towards a Dynamic Data Driven System for Structural and Material Health Monitoring. International Conference on Computational Science, **3**, (2006), 456–464.
2. Michopoulos, J., Tsompanopoulou, P., Houstis, E., Rice, J., Farhat, C., Lesoinne, M., Lechenault, F., DDEMA: A Data Driven Environment for Multiphysics Applications. In: Proceedings of International Conference of Computational Science - ICCS'03, Sloot, P.M.A., et al. (Eds.) Melbourne Australia, June 2-4, **LNCS 2660**, Part IV, Springer-Verlag, Haidelberg, (2003), 309–318.
3. Michopoulos, J., Tsompanopoulou, P., Houstis, E., Farhat, C., Lesoinne, M., Rice, J., Joshi, A., On a Data Driven Environment for Multiphysics Applications. Fut. Generation Comp. Sys., **21**:6, (2005), 953–968.
4. Farhat C., Chandesris M., Time-Decomposed Parallel Time-Integrators: Theory and Feasability Studies for Fluid, Structure, and Fluid-Structure Applications. Internat. J. Numer. Meths. Engrg, **58**:9, (2003), 1397–1434.

5. Farhat, C., Cortial, J., Dastillung, C., Bavestrello, H., Time-Parallel Implicit Integrators for the Near-Real-Time Prediction of Linear Structural Dynamic Responses. Internat. J. Numer. Meths. Engrg, **67**:5, (2006), 697–724.
6. Farhat, C., Hemez, F., Updating Finite Element Dynamic Models Using an Element-by-Element Sensitivity Methodology. AIAA J., **31**:9, (1993), 1702–1711.
7. Doebling, S.W., Hemez, F.M., Peterson, L.D., Farhat, C., Improved Damage Location Accuracy Using Strain Energy-Based Mode Selection Criteria. AIAA J., **35**:4, (1997), 693–699.
8. Donoho, D., Compressed Sensing. IEEE Transactions on Information Theory, **52**:(4), (2006), 1289–1306.
9. Farhat, C., Geuzaine, P., Brown, G., Application of a Three-Field Nonlinear Fluid-Structure Formulation to the Prediction of the Aeroelastic Parameters of an F-16 Fighter. Comput. & Fluids, **32**, (2003), 3–29.
10. Geuzaine, P., Brown, G., Harris, C., Farhat, C., Aeroelastic Dynamic Analysis of a Full F-16 Configuration for Various Flight Conditions. AIAA J., **41**, (2003), 363–371.
11. Lieu, T., Farhat, C., Lesoinne, M., Reduced-Order Fluid/Structure Modeling of a Complete Aircraft Configuration, Comput. Meths. Appl. Mech. Engrg, **195**, (2006), 5730–5742.