NONLINEAR DYNAMIC ANALYSIS OF STRUCTURES USING CELLULAR AUTOMATA

E. Salajegheh[1] ,S. Ghaderi Sohi[2] and A. Mohammadi[3]

[1] *Professor, Dept. of Civil Engineering , Shahid Bahonar Univesity, Kerman, Iran*
[2] *Graduate Research Assistant, Dept. of Civil Engineering, Shahid Bahonar University, Kerman, Iran*
[3] *Graduate Research Assistant, Dept. of Civil Engineering, Islamic Azad University, Kerman, Iran*
Email: eysasala@mail.uk.ac.ir, s.ghaderi@mail.uk.ac.ir, afsh_mohammadi@yahoo.com

ABSTRACT:

Emerging idea of Parallel Computing makes the use of new algorithms inevitable to deal with the problem of structural optimization. Simultaneous analysis and design has been a well known optimization method whose efficiency can be increased by the use of parallel computation methods. Cellular Automata are among those algorithms which their models are composed of regular lattice of cell or automata. Each cell may change its state at discrete, fixed times (iterations) according to a local rule based on behavior of the system which should be modeled. So the behavior of a system can be modeled by simple interaction of cells. This study contributes to nonlinear dynamic analysis of multi degrees of freedom (MDF) structures by Cellular Automata. The proposed method establishes local update rules which act on the stories of the structures as system cells. Any of the stories interacts with its neighbors, while applying a ground motion record to the structure base. This leads to the global dynamic response of structure after some iteration. The analysis update rules tend to solve the global differential equation of motion by solving some local differential equations at the cell levels. An outstanding feature of the proposed method is that the analysis of the structure is performed by solving some simple equations corresponding to single degree of freedom (SDF) systems. This scheme of the method is astonishing, as previous studies on training artificial neural networks for optimizing dynamic behaviors of structures can be extended to nonlinear phase. Another aspect of Cellular Automata which makes them efficient computational tools is their capability to be utilized efficiently by parallel processors for modeling complex systems.

KEYWORDS: Cellular Automata, nonlinear dynamic analysis, optimization, parallel computing

1. INTRODUCTION

Almost all methods which exist for the solution of coupled equations of motion for an MDF system subjected to dynamic forces can be categorized into two classes: direct integration and the modal superposition. In direct integration approach, the coupled equations are integrated using a numerical method. The word direct is used to imply that the equations are not transformed to another form before integrating. Modal superposition, on the other hand, is based on the idea of transforming the coupled equations to modal coordinates, where the displacement vector of an MDF system can be expanded in terms of modal contributions. Although it is an efficient method with a reasonable computational cost, it is incapable of considering nonlinear effects.

In recent decades that the size and complexity of problems in all fields of knowledge as well as structural analysis have been increased, production of more effective processors is inevitable. Using new developments in the field of semi conductors, computer industries advanced the computational capability of their processors, but this capability has an upper limit, and this limit is the motive for the idea of parallel computing. Today computer hardware engineers try to advance their processors by adding the ability of parallel processing. Massive Parallel Processors (MPP) are the result of the efforts in this field. Having this kind of processors, the next step is to develop algorithms for parallelizing the conventional methods used for solving different problems.

Traditional approach for structural analysis and design utilizes finite element based numerical analysis programs. While this approach works well for many problems, it does not parallelize efficiently on massively

parallel processors (MPPs), thus limiting the size and complexity of the problems which could be analyzed. New approaches are therefore needed. These methods need not to be faster than the conventional methods, as those do not utilize all the machine resources because of their serial nature. The aim of parallel processing is to discretize the problem into smaller portions which can be solved simultaneously by the parallel processor.

In the field of structural dynamic analysis the concept of parallel computation can best be implemented to modal superposition methods, as several similar computations are carried out for each independent mode of oscillation. The problem arises when nonlinear effects shall be considered. Direct integration, on the other hand does not have a potential of parallelizing. Cellular Automata paradigm, mostly used as an optimization tool in Civil Engineering, by its massive parallel computation capability can best be implemented for dynamic analysis of MDF structures. This approach, not only parallelize the conventional methods of analysis but also gives the optimizers an efficient medium for optimizing dynamic systems through the use of simultaneous analysis and design (SAND) approach, which analysis and design are combined in a single optimization problem. This is done by adding analysis unknowns such as displacement to design variables (Haftka 1989).Cellular Automata approach, unlike finite element method tries to solve the governing differential equation of a system locally rather than globally. The global behavior is thus modeled by local interaction of system components. Another interesting feature of using Cellular Automata for dynamic analysis is the possibility of training artificial neural networks as an alternative for exact local analysis procedure especially when nonlinear behavior is modeled.

2. CELLUAR AUTOMATA PARADIGM

The essence of Cellular Automata is the attempt to solve non-linear partial differential equations of complex system in an iterative localized manner. Cellular Automata tiles a problem domain into cells of equal size. Each cell in the lattice has the same set of simple rules that dictate how it behaves and interacts with its neighboring cells. The principle is that an overall global behavior can be computed by a group of cells that only know local conditions (Wolfarm 1994). If each cell only needs to know local conditions, then this minimizes the communication requirements and therefore the problem scales well on MPP (Slotta 2001). A cellular automaton is a discrete dynamical system. Each cell is a fixed point in a regular lattice. The state of each cell is updated at discrete time steps, based upon conditions in previous time steps. All the cells are updated every time step based on a local rule, thus the state of the entire lattice is updated every time step. In general, Cellular Automata are used to simulate the dynamic behavior of physical systems, and have been used successfully to represent a variety of phenomena such as diffusion of gaseous systems, solidification and crystal growth in solids, hydrodynamic flow and turbulence (Slotta 2001).

2.1. Cellular Automata Lattice

Form of the cellular space directly reflects physical dimensions of the problem being solved. Two sample lattice structures, representing one- and two-dimensional cellular spaces are shown in Figure 1, where cell locations are indicated by open circles. A three-dimensional space can be constructed by layering several of the two-dimensional ones, spaced equally so that the distance between them is the same as the distance between the cells in the plane. The lattice structures, however, are not limited to the rectangular ones, shown in the figure.

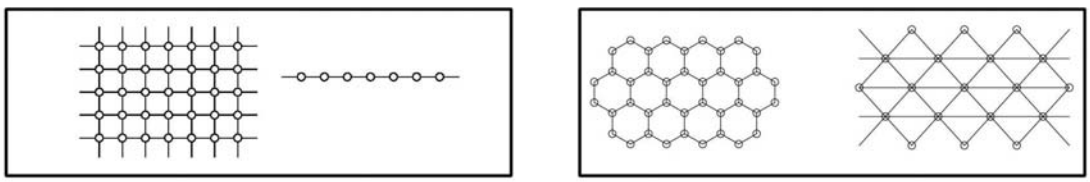

Figure1 Cellular Automata lattices

Cellular Automata based on other lattice systems such as two-dimensional trigonal and hexagonal lattices are also possible (Abdalla 2004). Each cell of the lattice has a value or set of values which may be binary, discrete

or continuous depending on the behavior which is to be modeled. They are updated over the course of iterations.

2.2. The Neighborhood

One of the most important features of Cellular Automata is the neighborhood structure. For updating the value of a cell, its own value and the values of neighboring cells should be considered. Configuration of the neighborhood structure is highly problem dependent and depends on the nature of the physical phenomenon that should be modeled (Abdalla 2004). Some common examples of neighborhood structures are shown in Figure 2. The cell to be updated is labeled as C, and the adjacent ones are labeled with letters representing the East, West, North, and South directions. Again, these are not the only neighborhood structures.

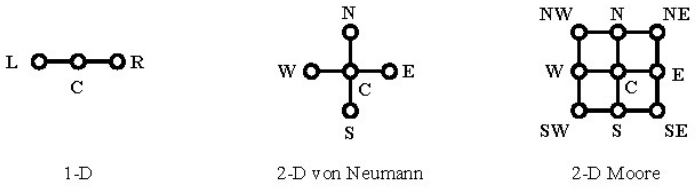

Figure 2 Cellular Automata neighborhood

2.3. Boundaries

As all cells have the same neighborhood structures, the problem arises for the cells at the boundaries of the system where a number of neighboring cells are out of the domain. A possible solution is to assume that border cells are connected to the cells on the opposite boundary as neighbors, forming a closed domain. For example, for a two-dimensional rectangular domain, a site on the left border has the site in the same row on the right border as its left neighbor. With the same update rule applied to all the cells, this yields what is called a periodic boundary condition which is representative of an infinite system. The type of the boundary condition to be used in a simulation depends on the physical application under consideration. Other types of boundary conditions may be modeled by using preset values of the cell for the boundary nodes or writing unique update rules for the cells at the boundary which provides substantial flexibility in introducing boundary conditions (Abdalla 2004).

2.4. CA Update Rules

To update the values of cells in any iteration some local update rules are which are directly dependent on the behavior of the system that is modeled. For example if the structural analysis of a system should be carried out, the rules are based on the minimization of the total potential energy at the cell level. In a computer implementation, the update rules that are applied to every cell of the lattice are like subroutine functions. The arguments for the function are the values of neighborhood sites, and the value returned by the function is the new value of the cell at which the function is being applied. For example, for the von Neumann neighborhood, the function has 5 arguments, f(C,E,W,N,S), which are the values of neighbors at iteration (t). It returns the value of the site C at the iteration (t+1).This is the key function of Cellular Automata which shows its inherent parallelism. As each cell has its own update rules which depends only on the values of its neighbors. It is conceivable that by assigning a simple processor to every so many cells of a large system of cells, one can increase the detail or the size of the system without increasing the time it takes to update the entire system. Thus, Cellular Automata simulations are highly suited for massively parallel computers. Although the update rules are usually the same for all cells in a domain, this is not always the case. Different analysis rules can be used in different parts of the domain. This is the case in SAND approach where usually there are analysis and design update rules.

3. APPLICATIONS OF CELLULAR AUTOMATA IN CIVIL ENGINEERING

Despite being introduced in the late 1940s, Cellular Automata paradigm is a newcomer in civil engineering and

this study is its second application in the field of dynamic analysis. The authors previously utilized it for linear dynamic analysis of structures (Salajegheh 2007). Biondini *et al.* carried out the durability analysis of concrete structures using Cellular Automata approach (Biondini 2004). Kicinger et al. represented Cellular Automata in topological structural design (Kicinger 2004). In the field of structural analysis Gurdal & Tatting introduced novel applications of Cellular Automata (Gurdal and Tatting 2000). Considering its evolutionary nature, they utilized it for simultaneous analysis and design of structures which is an effective tool for optimization. In combining analysis and design, Cellular Automata usually use fully stressed methodology in design phase which is incapable of considering optimization constraints. Salajegheh *et al.* optimized double layer space structures with the SAND approach by locally combining Cellular Automata and genetic algorithms (Salajegheh 2007).

4. PROBLEM DEFINITION

The aim of this study is to perform nonlinear dynamic analysis of MDF structures against seismic excitation, by means of Cellular Automata paradigm. Minimization of the total potential energy in a system leads to equilibrium equations. If this has been done globally, the finite element formulation of the structure is acquired. In Cellular Automata approach energy minimization and the derivation of equilibrium equations should be performed locally. Global equilibrium of the structure is then guaranteed by interactions between the local equilibrium equations. This is an iteration based method in which the structure gradually seeks its stable state.

Although dynamic analysis by Cellular Automata is not restricted to MDF shear structures, regarding its simplicity for acquiring equilibrium equation at cell level, a typical MDF structure with rigid beams is chosen. A 1-D cellular lattice is considered in which each floor in the structure represents a cell with two neighbors (upper and lower stories) (Figure 3). Each cell has its own specification such as the story mass, displacements, stiffness and damping of neighboring stories and itself. Therefore the state of a cell at the n^{th} iteration can be implied as:

$$S_n = \{m_i, U_i, (k_i, k_{i+1}), (c_i, c_{i+1})\} \tag{4.1}$$

Where m_i, U_i, k_i *and* c_i are mass, displacement, stiffness, and damping coefficients of the i^{th} story and S_n is the state of a cell at the n^{th} iteration. Knowing the state of all cells in the n^{th} time segment, some update rules should be derived using local equilibrium of cells to obtain the $n+1^{th}$ segment condition.

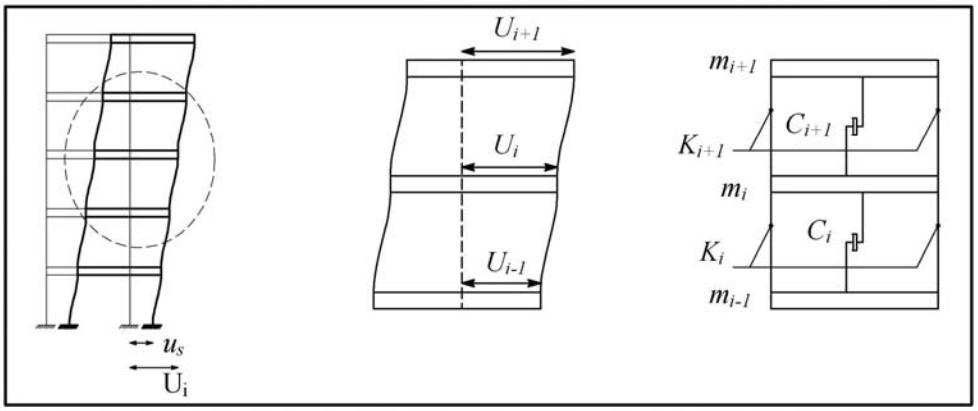

Figure 3 Configuration of Cellular Automata

4.1. Analysis Update Rule

Although there is no restriction on how damping is introduced to the model, for the consistency of the update rule and the numerical example (Chopra, 2005), which utilizes Rayleigh damping, Mass and stiffness proportional damping (Figure 4) are used.

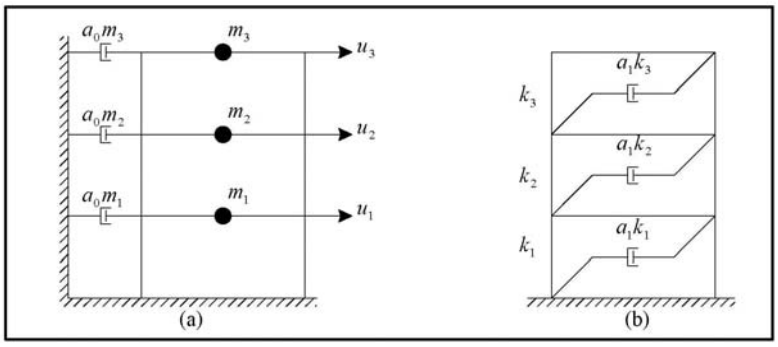

Figure 4 (a) Mass-proportional damping; (b) stiffness proportional damping

Free body diagram of a cell under the applied load from its neighbors is shown in Figure 5. The equation of motion is gained using D'Alembert's principle of dynamic equilibrium.

$$m_i\ddot{U}_i + k_i\left(U_i - U_{i-1}\right) + a_1k_i\left(\dot{U}_i - \dot{U}_{i-1}\right) - k_{i+1}\left(U_{i+1} - U_i\right) - a_1k_{i+1}\left(\dot{U}_{i+1} - \dot{U}_i\right) + a_0m_i\dot{u}_i = 0 \qquad (4.1.1)$$

The displacement relative to the base is implied by:

$$U_i = u_i + u_s \qquad (4.1.2)$$

Where u_i and u_s are relative displacement of i[th] story to the ground and base displacement respectively.

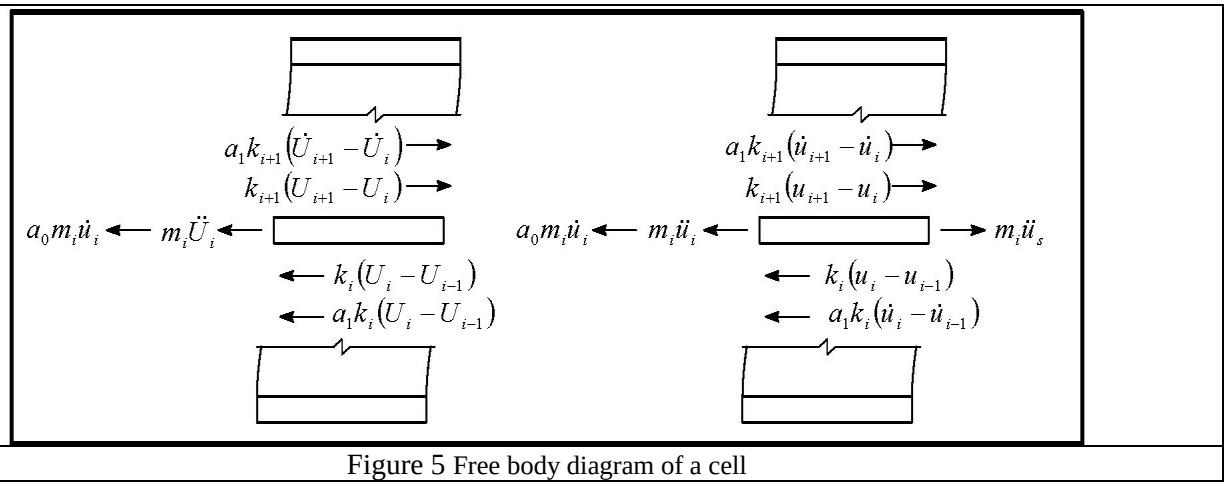

Figure 5 Free body diagram of a cell

Thus Eqn. 4.1.1 takes the form:

$$m_i\ddot{u}_i + k_i\left(u_i - u_{i-1}\right) + a_1k_i\left(\dot{u}_i - \dot{u}_{i-1}\right) - k_{i+1}\left(u_{i+1} - u_i\right) - a_1k_{i+1}\left(\dot{u}_{i+1} - \dot{u}_i\right) + a_0m_i\dot{u}_i = -m_i\ddot{u}_s \qquad (4.1.3)$$

Rewriting Eqn. 4.1.3, differential equation of motion for a cell takes the form:

$$m_i\ddot{u}_i + \left(a_0m_i + a_1k_i\right)\dot{u}_i + k_iu_i = -m_i\ddot{u}_s + k_{i+1}\left(u_{i+1} - u_i\right) + a_1k_{i+1}\left(\dot{u}_{i+1} - \dot{u}_i\right) + k_iu_{i-1} + a_1k_i\dot{u}_{i-1} \qquad (4.1.4)$$

This is the equation of motion for an SDF system which is subjected to a force defined by the right side of Eqn. 4.1.4, called the pseudo force, *P(t)*. Eqn. 4.1.4 has been used for dynamic analysis of linear systems (Salajegheh, 2007), but it's obvious that when nonlinear behavior is to be modeled it is of no use because in such a case, the last two terms on the right side of equation increase without bound, when columns of i^{th} story are yielded. In this study a change of variable is suggested as follows:

$$u_i' = u_i - u_{i-1} \tag{4.1.5}$$

For each cell it can be assumed that the displacement of the story relative to its bottom story (u_i') is to be calculated. In other word ground level is transformed to the bottom level of cell and the acceleration of ground and the $i-1^{th}$ story is algebraically summed. By this assumption the last two terms of Eqn. 4.1.4 are omitted and it takes the following form:

$$m_i\ddot{u}_i' + (a_0 m_i + a_1 k_i)\dot{u}_i' + k_i u_i' = -m_i(\ddot{u}_s + \ddot{u}_{i-1}) + k_{i+1}(u_{i+1}' - u_i') + a_1 k_{i+1}(\dot{u}_{i+1}' - \dot{u}_i') \tag{4.1.6}$$

$$P(t) = -m_i\ddot{u}_s + k_{i+1}(u_{i+1} - u_i) + a_1 k_{i+1}(\dot{u}_{i+1} - \dot{u}_i) + k_i u_{i-1} + a_1 k_i \dot{u}_{i-1} \tag{4.1.7}$$

Now the pseudo-force can be calculated easily, even in case of yielding if the elastoplastic relation between displacement and story shear is known. Considering the state of each cell at any iteration, *P(t)* is known and the differential equation of SDF system can be solved by standard methods of structural dynamics. By calculating u_i', u_i is easily estimated by Eqn. 4.1.5. This is an iterative procedure. For any u_i calculated in an iteration a new value for *P(t)* is derived. The process goes on until the convergence criteria would be satisfied.

4.2. Analysis Update Approach

Different approaches could be used in applying analysis updates rules to the cells for estimating the response of a structure against seismic excitation. One may seek total equilibrium at each time segment of ground motion record and then goes to other segments. This means that Cellular Automata iterations are performed at first segment of the record until the convergence criteria are met and then other time segments are considered. The other method is to seek equilibrium of all cells simultaneously all trough the record. In this way a cell displacements in all time segments, regarding neighboring cells information are calculated. Iterations go on until the steady state of all cells gained simultaneously. Although the second method is more time consuming but it has the advantage of proper parallelization and also useful in training artificial neural networks which are of great interest in the field of structural optimization.

5. NUMERICAL EXAMPLE

To illustrate accuracy of the proposed method a simple example is considered (Chopra, 2005). It is a five story building (Figure 6.a) idealized as a shear frame with elastoplastic relation between the shear force and story drift (Figure 6.d). The structure is subjected to El Centro ground motion. The average acceleration method is used to determine response equation 4.1.6 for each cell. Story stiffness and yield strengths which are normalized relative to the total weight of the structure are shown in Figure 6.b and Figure 6.c respectively.

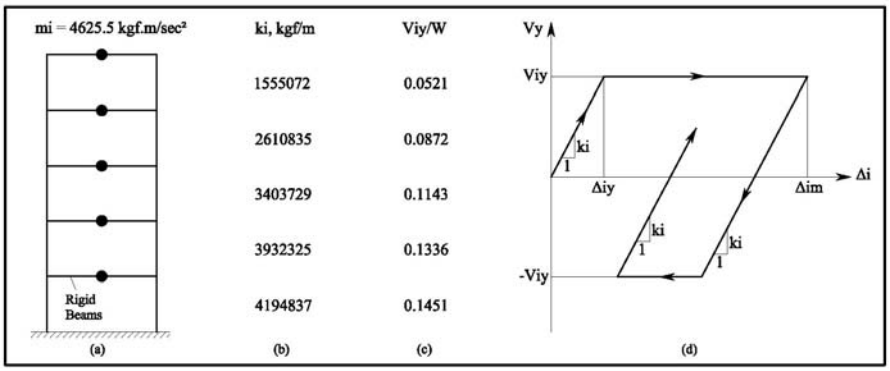

Figure 6 (a) system properties, (b)story stiffnesses, (c)story yield strengths, (d)elastoplastic relation between story shear and story drift

Response result, which is the displacement of top floor for the first 15 seconds of El Centro ground motion, gained by Cellular Automata methodology is presented in Figure 7. Corresponding result of the reference example (Chopra, 2005) is shown in Figure 8.

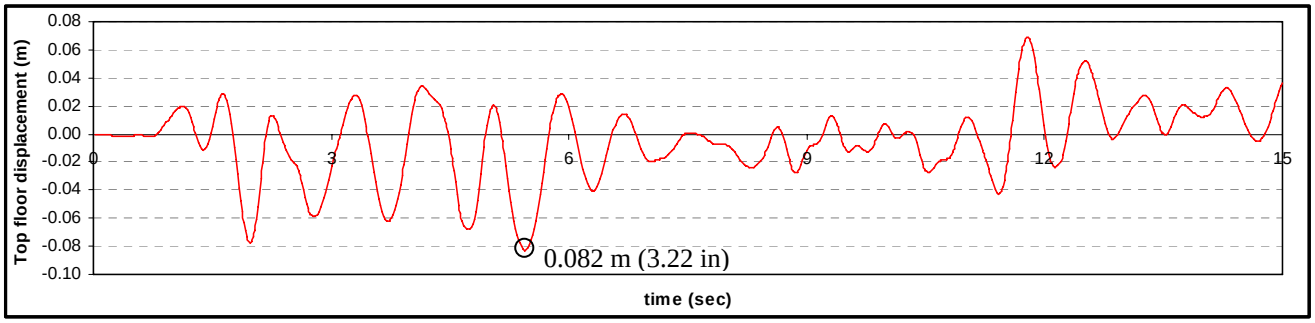

Figure 7 Calculated response by Cellular Automata methodology

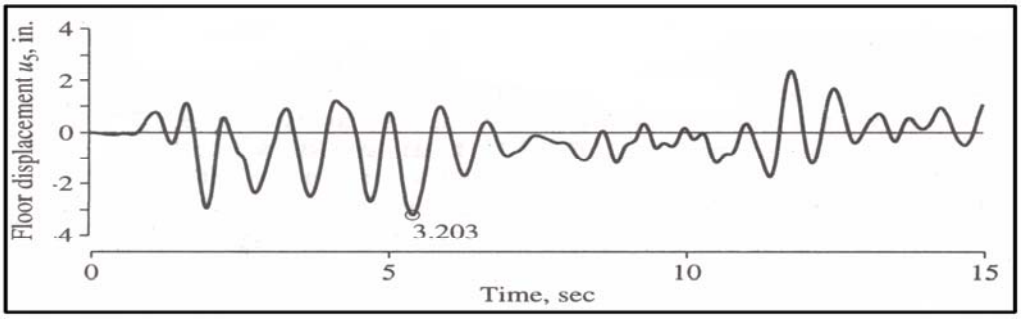

Figure 8 Actual response result

The displacement trace of building roof is in good compliance with actual results of the example. This is also true for the maximum displacement of roof which is estimated to be 0.082 m (3.22 in.) by CA method. The slight differences in results are due to the sensitivity of nonlinear analysis to parameters such as overshooting tolerance etc. The relation between story shear V_5 and story drift Δ_5, calculated by the proposed method in this study is shown in Figure 9.

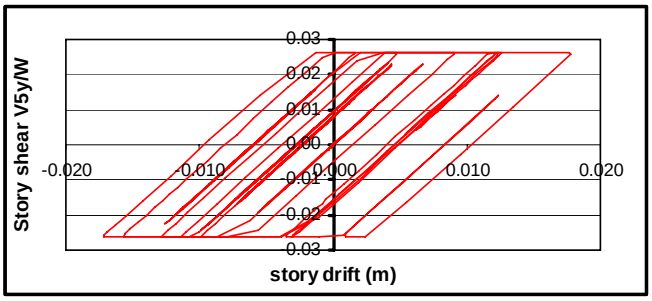

Figure 9 Story shear vs. drift for the fifth story

6. CONCLUSIONS

Dynamic analysis of MDF structures needs the solution of a set of coupled differential equation which should be solved simultaneously. This is impractical when degrees of freedom increase. Methods exist for the decoupling of these equations like modal superposition. Cellular Automata paradigm is in fact a numerical approach to solve sets of equations simultaneously. Previous studies have shown the efficiency of the method in structural analysis for static load cases where solving sets of equations derived from local analysis of cells was equivalent to the global solution of the structure finite element formulation.

This study shows the capability of Cellular Automata in estimating dynamic demands of structures especially when nonlinear behavior is to be modeled. Regardless of the type of structure, local formulation of the equation of motion leads to a set of differential equations representing the motion of SDF systems under dynamic loading. This is the interesting feature of Cellular Automata applications, because unlike the modal superposition, SDF systems derived from decoupling the set of coupled equations, should not be considered only in the elastic range. The fact that MDF structures can be analyzed through a set of SDF models is useful not only in the field of parallel computing, but also in optimization. Dynamic analysis has always been the challenge of optimizers, as it's a time consuming, costly procedure. Approximation of dynamic responses is a common practice in optimization methodology. Artificial neural networks are suitable for this purpose. If neural networks could be trained for the estimation of SDF structures response, Cellular Automata can easily be used to approximate the global behavior of a system under dynamic loading. Training neural networks for dynamic analysis of SDF systems have been carried out previously by Salajegheh et al. (Salajegheh 2004). The trained network was combined with modal superposition analysis to estimate dynamic response of elastic structures. With the use of the methodology presented in this study the future goal is to train neural networks which are capable of considering nonlinear responses.

REFERENCES

Abdalla, M.M. (2004). Applications of the Cellular Automata Paradigm in Structural Analysis and Design, *M.Sc. Thesis*, Cairo University, Cairo, Egypt.

Biondini, F., Bontempi, F., Frangopol, M.D. and Malerba, G.P. (2004). Cellular Automata Approach to Durability Analysis of Concrete Structures in Aggressive Environments. *Journal of Structural Engineering ASCE* 130(11), 1724-1737.

Chopra, A.K. (2005). Dynamics of Structures Theory and Applications to Earthquake Engineering, Prentice-Hall India.

G¨urdal, Z. and Tatting, B. (2000). Cellular Automata for Design of Truss Structures with Linear and Nonlinear Response. *Proc. 41st AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conf., AIAA Paper 2000-1580* Atlanta, GA.

Haftka, R.T., Kamat, M.P. (1989). Simultaneous Nonlinear Structural Analysis and Design. *Computational Mechanics* 4, 409-416.

Kicinger, R., Arciszewski, T., and De Jong, K.A. (2004). Morphogenic Evolutionary Design: Cellular Automata Representations in Topological Structural Design. *Adaptive Computing in Design and Manufacture VI, IC Parmee, ed., Springer-Verlag*, 25-38, London , U.K.

Salajegheh, E., Heidari, A. (2004). Optimum Design of Structures against Earthquake by Wavelet Neural Network and Filter Banks. *Earthquake Engineering and Structural Dynamics* 34, 67–82.

Salajegheh, E., Ghaderi Sohi, S., Mohammadi, A. (2007). Optimization of Double Layer Space Structures with the Combination of Cellular Automata and Genetic Algorithm, *Proc. of the 5th International Congress of Civil Engineering*, Tabriz, Iran.

Salajegheh, E., Ghaderi Sohi, S., Mohammadi, A. (2007). A New Method for Dynamic Analysis of Structures Using Cellular Automata. *Proc. of International Conference on Seismology and Earthquake Engineering*, Tehran, Iran.

Slotta, J.D. (2001). Structural Design Using Cellular Automata. *M .Sc. Thesis*, Virginia Polytechnic Institute and State University, Virginia, U.S.

Wolfram, S. (1994). Cellular Automata and Complexity. Collected Papers, Addison-Wesley, Reading, MA.