A Constant Displacement Iteration Algorithm for Nonlinear Static Push-Over Analyses

Graham C. Archer

Assistant Professor, School of Civil Engineering, University of Purdue
1284 Civil Engineering Building, West Lafayette, IN 47907-1284, USA.
Email: archer@purdue.edu

ABSTRACT

This paper presents the formulation and implementation of a new static push-over analysis algorithm for the seismic rehabilitation of building frames, in accordance with "NEHRP Guidelines for the Seismic Rehabilitation of Buildings". The concept of the push-over analysis is described and problems associated with its implementation in a nonlinear finite element program are examined. Earlier work, which contributed to the development of the strategy, is reviewed. The theory of the solution algorithm is presented with special consideration placed on the stability of the algorithm and the method of iteration for the nonlinear behavior. Examples are carried out that demonstrate the remarkably stable and efficient procedure.

KEYWORDS

1. Introduction

The current engineering practice for the seismic analysis of building frames is moving away from simplified linear-elastic methods of analysis, and towards a more complex nonlinear-inelastic technique. A reflection of this is contained in the "NEHRP Guidelines for the Seismic Rehabilitation of Buildings" (FEMA Publication 273 [1,2]). FEMA 273 serves as a foundation for the future development and implementation of building code provisions and standards. These provisions focus on the nonlinear behavior of structural response and employ procedures not previously emphasized in seismic codes.

FEMA 273 introduces the Nonlinear Static Procedure (NSP) for the modeling and analysis of seismically rehabilitated building frames. The NSP, also known as a push-over analysis, is a relatively simple way to explore the design of a structure. It consists of pushing a mathematical model of a building frame over to a prescribed displacement in order to predict the sequence of damage in the inelastic range of behavior. Unfortunately, the unique displacement-based nature of the push-over analysis makes it difficult to implement in general-purpose finite element programs. Specialized solution algorithms are required. This paper presents a simple and efficient algorithm for implementing the push-over analysis in a nonlinear finite element program. The validity of the algorithm is demonstrated through the analysis of nine steel frame buildings.

2. The nonlinear static procedure (NSP)

Under the NSP, a nonlinear-inelastic model of a building frame (Fig. 1), with gravity loads Q_G, is laterally loaded until either a predefined target displacement δ_t is met, or the model collapses. The target displacement represents the maximum expected displacement for the design earthquake. The control node, at which the target displacement is defined, is taken as the center of mass at roof level. Two vertical distributions of lateral load F are considered, a uniform

pattern and a modal pattern. The analysis provides a base shear versus control-node displacement relationship, which indicates the change in stiffness as well as lateral load capacity of a structure.
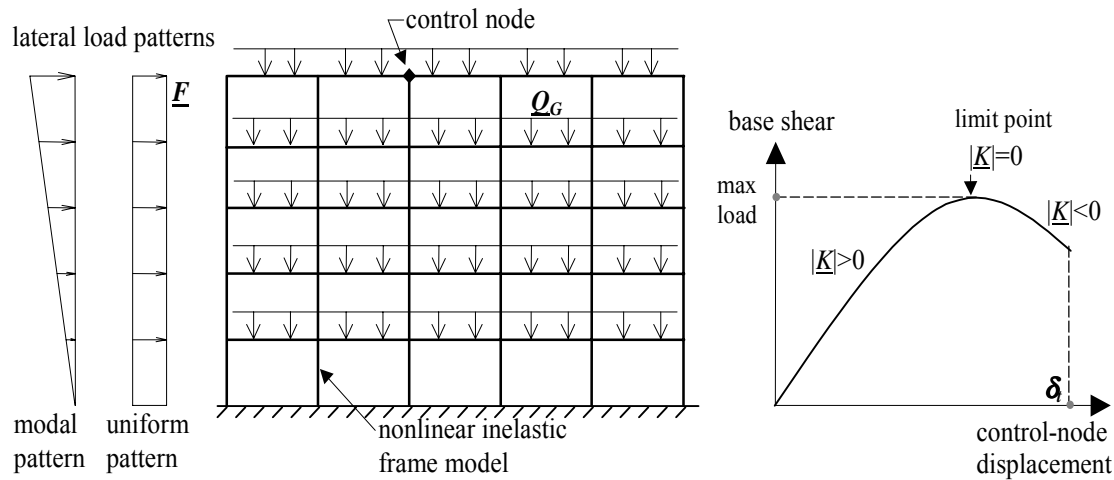

Figure 1: Nonlinear Static Procedure (push-over analysis).

3. Problem statement

A typical nonlinear static analysis consists of advancing the displacements Δu in response to changes in the applied loads ΔF. In general, this is accomplished by linearizing the stiffness K of the structure and in some manner solving

$$\Delta u = K^{-1} \Delta F \, , \tag{1}$$

until equilibrium between the applied loads and the structures resisting forces is achieved. Contrarily, in a push-over analysis the lateral load is not known in advance and can not be treated as an independent variable. In the NSP, a single displacement component is specified (at the control node) and the lateral loads must be scaled up or down in order to produce desired displacement. This unique scaling of the lateral loads, while holding the gravity loads fixed, requires special consideration in the development of an appropriate solution algorithm.

Additional problems are encountered with the NSP as the structure is taken to, and past, the point of maximum load. Prior to reaching the maximum load, the tangent stiffness of the model is positive-definite. That is, the determinate of the tangent stiffness matrix is greater than zero ($|K| > 0$). Thus, the structure is stable and a unique solution to equation (1) exists. For this portion of the analysis, any of the common iterative solution schemes (such as Newton-Raphson iteration) and numerical solvers (such as Choleski decomposition) would be appropriate.

Beyond the maximum load though, $|K| < 0$ and the structure is unstable. This introduces problems in both the solution scheme and in the numerical solution of equation (1). In order to increase the lateral displacements, the lateral loads must be decreased. This counter-intuitive behavior is not anticipated by most general solution schemes and as a result they fail to find a solution. Furthermore, the negative-definite stiffness matrix precludes the use of the efficient numerical solvers employed in most structural analysis programs.

At the point of maximum load though, there exists no unique solution to equation (1) since $|K| = 0$. Also, in the vicinity of the limit point, the results are inaccurate. Therefore, a solution scheme based on the tangent stiffness matrix, such as Newton-Raphson iteration, would fail at

the point of maximum load. Other linearization schemes, (such as Modified Newton-Raphson or Quasi-Newton iteration) are not immune to the problem since the possibility of a zero, or close to zero, determinate exists.

4. A review of displacement control methods

Since the NSP specifies a target displacement, not a target load, a displacement-based solution scheme fits naturally into the framework. Argyris [3] first presented the concept of incrementing displacements instead of loads in the development of an algorithm for large displacement analysis. Briefly, the technique determines the load in each incremental step based on the displacements obtained in the preceding analysis step. However, at the limit point this method requires an estimation of the structural response beforehand. Therefore, it does not necessarily guaranty the correct solution.

Pian and Tong [4] suggested a more general method. Their algorithm is primarily based on incrementing the load, until the limit point is reached. When the determinant of the matrix K changes the sign, i.e. at the limit point, the technique switches to increments of displacement. A single displacement component is then adopted as the new independent parameter and the equations of equilibrium (Eq.1) are partitioned and rearranged to isolate the unknowns. Unfortunately, the resulting system of equations is not symmetric, thus rendering the method difficult to implement and requiring significant computational effort to solve.

The general concept of controlling a single displacement component proposed by Pian and Tong was accepted and modified to fit into a 'standard' finite element program, by Zienkiewicz [5]. In Zienkiewicz's approach, a large fictitious spring is added to the model. The addition of the spring not only helps constrain the displacement of the model, but also helps eliminate the possibility of overall negative stiffness. The external loads are represented as a reference vector, F, and a scalar magnitude λ. Thus, the incremental equilibrium equation becomes:

$$K \, \Delta u = \Delta \lambda \, F \,, \tag{2}$$

In order to obtain $\Delta \lambda$, a single displacement component is prescribed at the spring location. The formulation for the constraint condition is established in two stages. In the first stage, only the displacement increment $\Delta \delta_c$ is applied to the structure (at the control node 'c') and the corresponding spring reaction R_c is obtained. In the second stage, the system acts under reference loads F with $\Delta \delta_c = 0$. The corresponding spring reaction R_F is obtained.   The superposition of the two stages that yields zero reaction in the spring defines the unknown load increment $\Delta \lambda$. Once, the load increment is determined, the analysis proceeds to the next incremental step. At each step, the structural stiffness matrix K is updated to account for the nonlinear behavior of the structure. Unfortunately, this simple incremental procedure results in an accumulation of equilibrium error that is mitigated only by greatly reducing the increment in displacement.

Sharifi and Popov [6] suggested an expansion on the concept of the fictitious spring. In a more general approach to ensuring that the stiffness matrix remains positive-definite throughout the range of analysis, they suggested the addition of number of fictitious springs. The mathematical formulation is similar to the Penalty Function method of constraint resolution and yields the external load scale factor λ as a function of nodal displacement increments, load vector and stiffness of a fictitious elastic spring. When large values are assigned to the springs, the resulting solution is displacement-controlled. However, the identification of the proper spring location and their appropriate values is unique to the model and not easy to determine.

Batoz and Dhatt [7] proposed an iterative procedure that corrects the equilibrium error in Zienkiewicz's approach. While using the same two-stage approach, their technique does not involve the addition of the fictitious spring. Thus, the constraint condition is expressed in terms

of displacements at the control node, rather than reactions. Within each incremental step, iterations are performed until the equilibrium unbalance is removed form the structure. In each iteration step, the value for $\Delta\lambda$ is improved until desired accuracy (such as minimum unbalance requirement) or the maximum number of iterations is achieved.

A review of nonlinear analysis strategies, written by Clarke and Hancock [8], also presents a modification to Zienkiewicz's approach termed "Iteration at constant displacement". This strategy is similar to Zienkiewicz's, in that a "key" displacement component is kept constant during the iterations. However, this is accomplished by proportionally adjusting the applied loads, not through the use of a large fictitious spring. Thus, unless extraordinary measures are taken, the strategy will fail at the point of maximum load ($|K| = 0$). In addition, since no action is taken to prevent a negative-definite stiffness matrix ($|K| < 0$), an appropriate (but less efficient) numerical solver must be chosen.

5. Problem solution

The primary drawback to displacement control techniques is the need to select a displacement component for the control of the iterative process. An appropriate choice requires prior knowledge of the structure's behavior and defies generalization. Furthermore, the maintenance of a positive-definite stiffness matrix throughout the analysis is difficult to assure. Due to this lack of generality, displacement control methods were mostly left forgotten and substituted by more promising schemes such as the Arc Length method and the Work Control method. These methods include all of the displacement components, rather than a single one, for controlling the direction of iteration.

However, the specified single-point displacement of the NSP, leads one to re-assess the displacement control techniques. Since the control node, at which the displacement is prescribed, is known in advance, the main drawback of displacement-controlled methods is eliminated. Additionally, the attachment of a single, grounded, horizontal spring to the control node to regulate displacements will also serve to "prop up" the model as it becomes globally unstable. Thus, the primary source of the zero or negative stiffness is eliminated.

Hence the proposed solution scheme for the NSP, is a variation on the Iteration at Constant Displacement strategy with the addition of Zienkiewicz's fictitious spring and a separation of the constant gravity load and variable lateral load. Briefly, the scheme consists of the following steps: First the gravity loads are applied to the model and held constant. A grounded horizontal spring is then attached to the control node. Lateral displacements are then applied at the control node in increments. At each increment, the lateral loads and the structure's state are iteratively adjusted until they are in equilibrium with each other. The procedure continues until the desired displacement at the control node is attained.

The details of the scheme are contained in the initial constraint formulation used to impose the increment in displacement at the control node, an iterative cycle to account for the structure's nonlinear-inelastic behavior, and the fictitious spring.

5.1 The initial constraint formulation

In a linear-elastic structure that is otherwise in equilibrium, the imposition of an increment in displacement at the control node causes a load unbalance. The purpose of the initial constraint formulation is to remove the unbalance. Consider the structure shown in Fig. 2a that is in equilibrium with the current external loads (lateral loads λF + gravity Q_G) and control node displacement δ_c. When the displacement increment $\Delta\delta_c$ is applied to the control node, a new equilibrium state will be obtained by incrementing the lateral loads (Fig. 2b). The scale factor $\Delta\lambda$ is the unknown parameter.
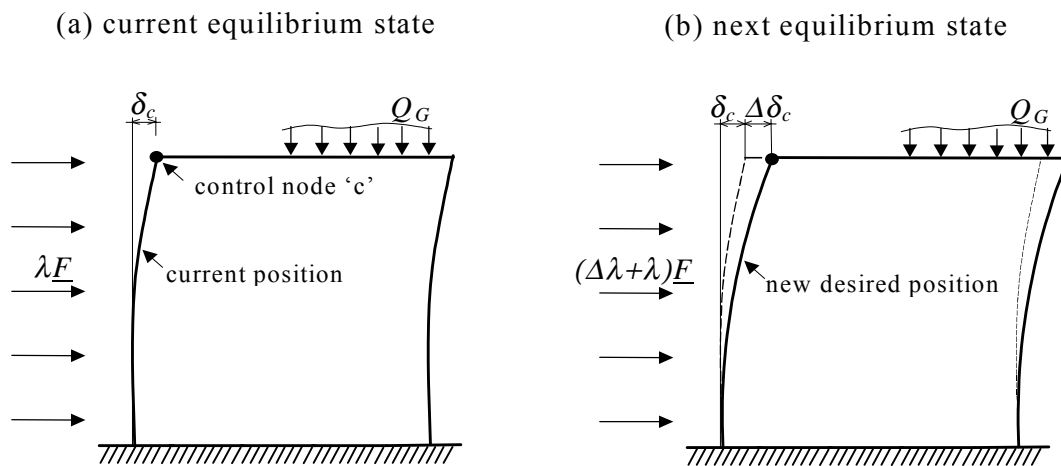
(a) current equilibrium state   (b) next equilibrium state

Figure 2. Adjacent equilibrium states.

The first step of the initial constraint formulation consists of applying an additional displacement increment $\Delta\delta_c$, at the control node. To maintain equilibrium in this new position, an unbalanced spring reaction R_U (Fig. 3a) is required at the control node. The second step of the formulation consists of applying the unit lateral load pattern F, by itself, to the current equilibrium state. To maintain the current control node displacement δ_c, another reactive force R_F (shown in the negative sense in Fig. 3b) is needed.

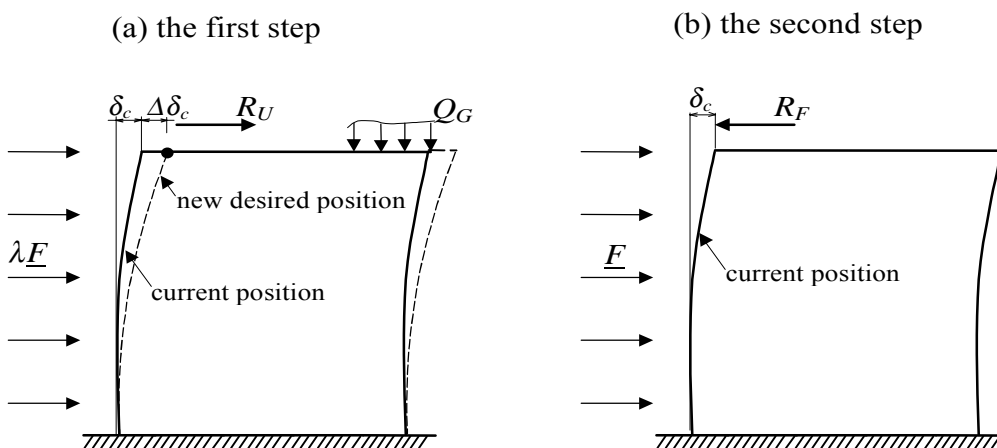(a) the first step   (b) the second step

Figure 3. Steps of the initial constraint formulation

If the structure were to remain linear-elastic, it would be a simple matter (from Zienkiewicz) to superimpose a portion of the second step onto the first, in order to remove the reactive forces at the control node. Using a scale factor $\Delta\lambda$, the reaction at the control node is defined by

$$R_U + \Delta\lambda R_F = 0 , \qquad (3)$$

Thus, we obtain:

$$\Delta\lambda = -\frac{R_U}{R_F} , \qquad (4)$$

In a stable, linear-elastic structure, R_F will be a negative number (reactive force to the left on Fig. 3b). Hence, $\Delta\lambda$ will be a positive number. If the structure were linear-elastic but unstable,

both R_F and R_U will be negative numbers. $\Delta\lambda$ will now be a negative number, thus indicating that a decrease in lateral load is necessary.

5.2 Iterative cycle

If structure were to remain linear-elastic, the application of equation (4) would predict the correct increase to the lateral loads necessary to hold the structure in the new displaced position. The increment in the displaced shape of the structure (with the control node held in the new position) can be calculated from equation (2) using the current linearized stiffness. The state of the structure is then updated. In a linear-elastic structure, this new state will be in equilibrium with the total lateral and gravity loads. For a nonlinear softening structure, $\Delta\lambda$ will be over-predicted, conversely, for a stiffening structure, $\Delta\lambda$ will be under-predicted. Thus, before proceeding to the next increment in displacement, the equilibrium error in the structure is eliminated through iteration.

Each iteration commences with a determination of the unbalanced load, i.e. the difference between the total applied load and the internal resisting force from the structure, and the current stiffness linearization. The unbalanced load is applied to the structure (with the control node held fixed) to produce the new unbalanced reaction at the control node, R_U. The new reaction force, R_F, due to the unit lateral load is recalculated for the current linearization. Equation (4) provides a new estimate for $\Delta\lambda$, which is used to recalculate the displaced shape and update the state of the structure. The iterations continue until the equilibrium error is acceptably small.

5.3 Fictitious spring

In the prior calculations, it was assumed that the control node can be held at the desired displacement, and that the reaction forces at the control node can be calculated. This is accomplished by the addition of a large fictitious spring and accompanying large fictitious force at the control node as shown on Figure 4(a). To dictate the motion of the node, the spring is assigned a very large stiffness value, K_s, equal to $10^5 \times \|K\|_\infty$. Thus, to produce a prescribed displacement δ_c at the control node 'c', a control force $F_s = \delta_c K_s$ is applied. Once a new control node increment $\Delta\delta_c$ is assigned, the control force is incremented by $\Delta\delta_c K_s$ (Figure 4(b)) and kept constant during that incremental step. The size of the force and spring stiffness, relative to the stiffness of the structure, guarantees the correct displacement of the control node, regardless of the actions of the structures.

Figure 4 represents equilibrium states. That is, structure states that require no additional reaction at the control node to produce the desired displacement. Therefore, the force in the spring is due only to the control force and consequently the compressive deformation of the spring precisely equals δ_c ($\varepsilon = 0$). For states that require a reaction at the control node to maintain structure equilibrium (for example the calculation of R_U and R_F) the structure will slightly lean, or pull, on the spring. Thus, the deformation of the spring will differ from the desired amount by ε (many orders of magnitude smaller than δ_c). Consequently, the reaction at the control node due to the equilibrium error in the structure is given as εK_s.

(a) current equilibrium state ($\varepsilon = 0$)   (b) new equilibrium state ($\varepsilon = 0$)
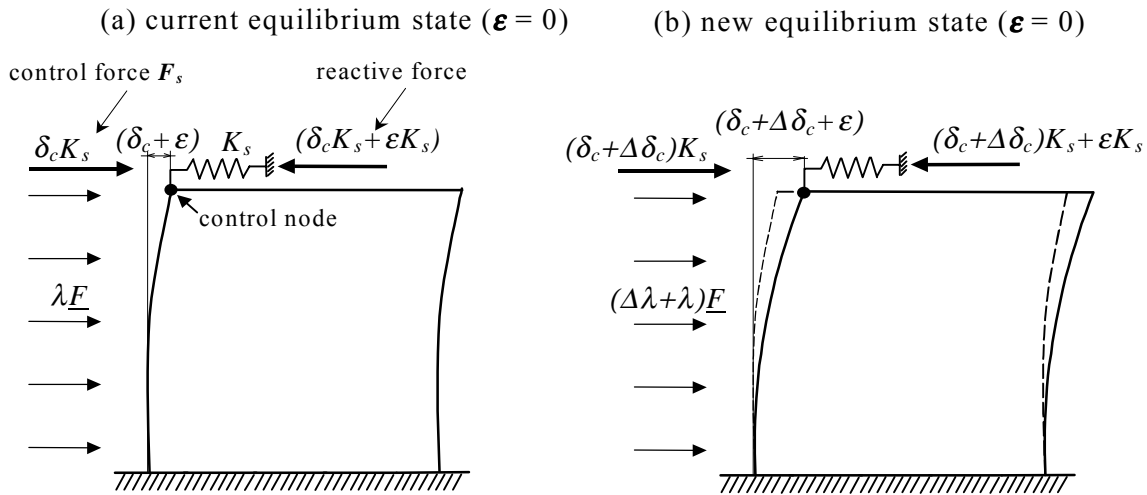

Figure 4. Control force.

6. A detailed description of the proposed algorithm

The previous formulations form the basis for the proposed NSP solution algorithm shown in Figure 5. Within each step, the initial stage is distinguished from the iterative scheme that follows.

Prior to the analysis, some initializations are necessary. The target displacement δ_t, lateral load pattern F, and gravity loads Q_G are known, stationary values. Once the structural stiffness matrix K is assembled, the fictitious spring stiffness K_s is determined and used as a constant throughout the analysis. When the structural stiffness matrix K is assembled, the spring stiffness K_s is added to the stiffness component in K at the control node. Finally, the scale factor λ is set to zero. This provides the initial value for external load-vector Q equal only to the gravity loads Q_G.

6.1 The first incremental step

The first incremental step is significantly different from the subsequent steps. It provides the analysis of the structure due only to the gravity loads. Before the first incremental step takes place, the initial values for control node displacement δ_c and its increment $\Delta\delta_c$ are set to zero, as are the nodal displacements of the structure u and the resisting force vector Q_R. Therefore, the structure is analyzed due only to the gravity loads Q_G. An iterative scheme is provided to account for any possible nonlinear behavior of the structure for these initial loads.

6.2 Successive steps

Each incremental step begins with an additional displacement increment $\Delta\delta_c$ applied to the control node (eq.5a). In the application of $\Delta\delta_c$, the external load component at the control node $Q\big|_c$ is updated by adding a force increment equal to $K_s(\Delta\delta_c)$ (eq.6). The unbalance U is introduced to the structure as the missing force at the control node. It is obtained from the difference between the current external load vector Q and the resisting force vector Q_R (eq.7). The magnitude of the unbalanced load must be compared to the specified convergence tolerance (eq.8). If the magnitude is smaller than a specified value, the analysis takes the next increment step. If the unbalance is larger than the tolerance, the iteration cycle commences.
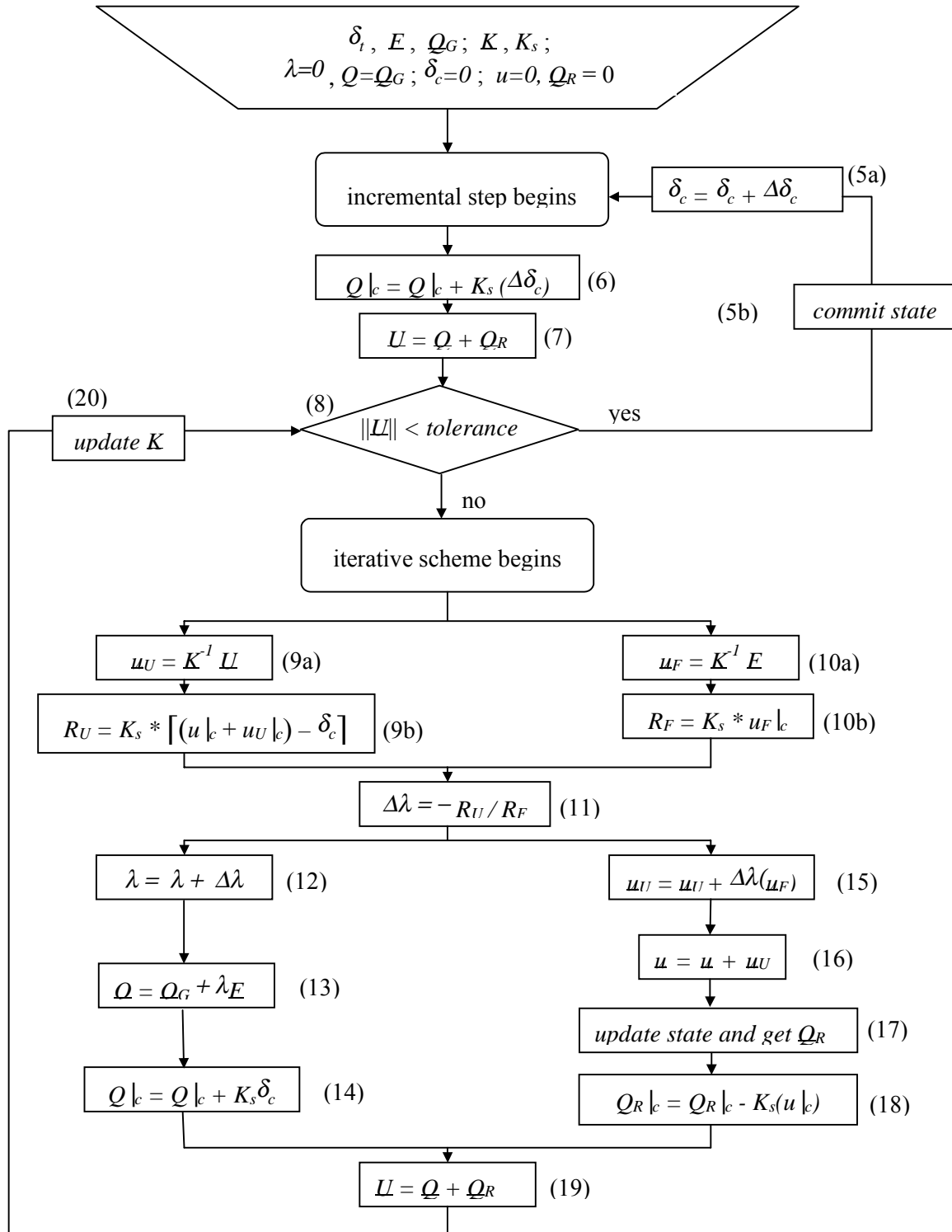
$$\delta_t \, , \; E \, , \; Q_G \, ; \; K \, , K_s \, ;$$
$$\lambda=0 \, , \; Q=Q_G \, ; \; \delta_c=0 \, ; \; u=0, \; Q_R=0$$

incremental step begins ← $\delta_c = \delta_c + \Delta\delta_c$ (5a)

$Q|_c = Q|_c + K_s \,(\Delta\delta_c)$ (6)

(5b)   *commit state*

$U = Q + Q_R$ (7)

(20)

update K ← (8) $\|U\| < tolerance$ yes

no

iterative scheme begins

$u_U = K^{-1} U$ (9a) $u_F = K^{-1} E$ (10a)

$R_U = K_s * \lceil (u|_c + u_U|_c) - \delta_c \rceil$ (9b) $R_F = K_s * u_F|_c$ (10b)

$\Delta\lambda = - R_U / R_F$ (11)

$\lambda = \lambda + \Delta\lambda$ (12) $u_U = u_U + \Delta\lambda(u_F)$ (15)

$u = u + u_U$ (16)

$Q = Q_G + \lambda E$ (13)

update state and get Q_R (17)

$Q|_c = Q|_c + K_s\delta_c$ (14) $Q_R|_c = Q_R|_c - K_s(u|_c)$ (18)

$U = Q + Q_R$ (19)

Figure 5. The Solution Algorithm

6.3 Iterative scheme

From the unbalanced load vector U, the corresponding displacements u_U are obtained (Eq. 9a) and the spring reaction R_U is calculated (Eq. 9b). This reaction arises as a result of the difference between the actual control node displacement and the prescribed node displacement δ_c, set for that increment step. The actual control node displacement is obtained by updating the

current displacement value $u\vert_c$ with the new unbalanced displacement $u_U\vert_c$. By applying the unit lateral load vector to the structure, the displacements u_F are obtained (Eq. 10a) with the corresponding spring reaction R_F (Eq. 10b).

The increment to the load scale factor $\Delta\lambda$ is obtained by seeking to eliminate the reaction force at the control node. It is calculated as the ratio of R_U and R_F (Eq. 11). It can be seen that if R_U and R_F are both in the same direction, then $\Delta\lambda$ will be negative and lateral load will be decreased. This will occur for a progressively softening system. On the other hand, if R_U and R_F are of the opposite sign, $\Delta\lambda$ will be positive and external load is increased. This will occur for a progressively stiffening system. The value $\Delta\lambda$ is then added to the previous value of λ (Eq.12).

The total load vector Q is equal to the sum of the gravity load Q_G and the lateral load λF. When the lateral load is scaled to the new value λF, the external load vector Q is updated (Eq.13). The additional fictitious force is then reinstated in Q (Eq. 14). The displacement increments, due to the increase of lateral load, are equal to $\Delta\lambda\ (u_F)$. These are added to unbalanced displacements u_U (Eq.15). The nodal displacements u are then corrected for the new unbalance u_U (Eq.16). The element states (strains and stresses) are updated and the new internal resisting force Q_R is obtained (Eq.17). The spring reaction $K_s\ (u\vert_c)$ is subtracted from the Q_R component at the control node $Q_R\vert_c$ (Eq.18). The new unbalance vector U is then determined (Eq.19) and the stiffness matrix K is updated before proceeding to the next step (Eq.20).

The unbalance is checked against the allowed tolerance to determine if the iterative scheme has converged. If the unbalance is larger than the tolerance, the analysis continues with the next iteration. The current values of δ_c, λ, u and U from the previous iteration are used in the next iteration step. The iterations will proceed until the unbalance is removed from the structure, or until the maximum number of iterations is reached. In the latter case, the incremental value for $\Delta\delta_c$ is decreased and the analysis repeats the incremental step.

If the iteration scheme has converged, the next incremental step takes place. The initial values for δ_c, λ, Q, Q_R and u are taken from the final values of the last iteration of the previous incremental step. The incremental steps proceed until the current control node displacement δ_c reaches the final target value δ_t.

6.4 Control of stability

The primary motivation for the addition of the fictitious spring at the control node is to aid in the removal of the overall zero and negative stiffness in the model. While this does control most numerical difficulties in the analysis, it does not eliminate them. Local failure mechanisms may still exist (Fig. 6b).

As shown in Fig. 6a, global type failure mechanisms are constrained by the addition of the spring at the control node. The global mechanism shown consists of plastic hinges at the ends of the beams and at the base of the first story columns. This represents a globally unstable structure with unrestrained lateral movement. The fictitious spring, attached to the top of the structure, provides the necessary lateral support. Thus, the mechanism is constrained and the analysis can proceed.

On the other hand, the addition of the spring does nothing to restrain the local mechanism shown in Fig. 6b. In this case, plastic hinges develop in the first floor and second floor columns. Thus, the second floor is unrestrained in the horizontal direction. Another typical mechanism occurs when all the beams and columns at a joint develop plastic hinges. The joint itself is then unrestrained for rotational movement.

Although the occurrence of a local mechanism in the analysis represents a serious deficiency in the structure, it does not at all represent a failure of the proposed solution scheme. In fact, one of the motivations of the NSP is for the analyst to identify the possible formation of these load-limiting mechanisms. Thus, the requirement on the analysis algorithm is that the local mechanism be identified and that its formation does not cause the algorithm to fail. Local mechanisms are easily identified in the decomposition of the stiffness matrix. To permit the analysis to proceed, a small, grounded spring (a stiffness of $10^{-12} \|K\|$ is sufficient) is added to the appropriate degree of freedom. The small spring eliminates the numerical difficulty of the zero stiffness, yet allows near unrestrained deformations as an indication to the analyst that a local mechanism has formed.
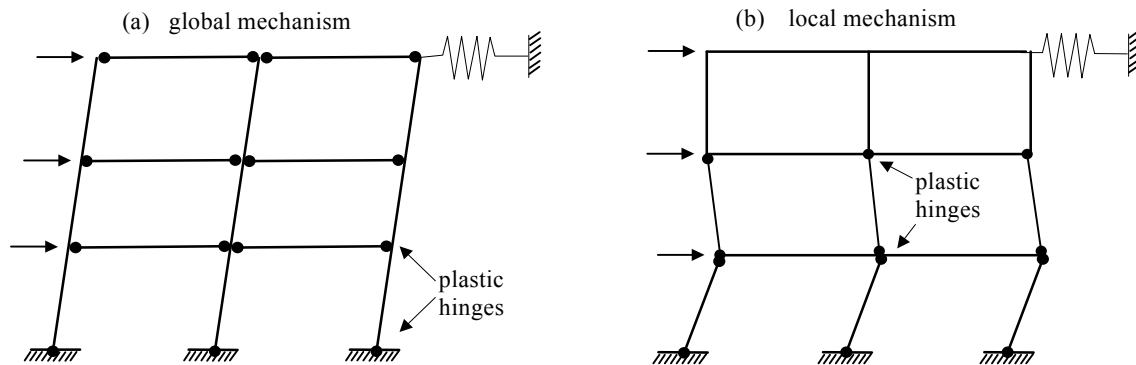

Figure 6. (a) a global mechanism is constrained and (b) a local mechanism is not caught.

6.5 Update-state and commit-state semantics

In the process of seeking equilibrium at the desired control node displacement, the model analysis will proceed through possibly many trial states. The model must be able to move between the trial states without affecting its ability to reach the correct equilibrium state.

One solution for this is to demand that the structure remain elastic (but not necessarily linear) during the state updates (Eq. 17). Unrecoverable damage is permitted only when a valid is committed (Eq. 5b). This approach works fairly well, but can fail when the true path from one equilibrium state to another demands unrecoverable damage.

Consider the column with elastic-perfectly-plastic end hinges bent in double curvature as shown in Fig. 7. In state 1 (Fig. 7a, the lower hinge has just yielded. A single curvature displacement increment (Figs. 7b and 7d) is to be added to the external nodes A and D, in order to enable the upper hinge to yield. The state of the column is then re-evaluated. The application of half the desired displacement increment (Fig. 7b) produces state 2 (Fig. 7c). Due to the flexibility of the lower hinge, the rotation distributes such that the lower hinge continues to plastically rotate, while the upper hinge opens elastically until the limit state is reached. Further displacement (Fig. 7d) causes the upper hinge to open plastically. However, the lower hinge now reverses its motion and begins to open elastically. The lower hinge is now on a different equilibrium path leading to state 3 (Fig. 7e).

Thus, to move between state 1 and state 3, the lower hinge must accommodate unrecoverable damage. That is, if the displacement increment were reversed, the column would not return to state 1. Hence, it is not sufficient to merely demand that the structure remain elastic during a state update (Eq. 17). If the structure had to remain elastic, then at state 2, the lower hinge could not close elastically. The resisting force from the column would then be in error. This would result in the distribution of the erroneous force elsewhere in the structure.

The correct interpretation of a state update (Eq. 17) is that each update must proceed from the last committed state, regardless of last trial state. Unrecoverable damage is permitted during the update. Since the last committed state is saved, the unrecoverable damage from an incorrect trial state is simply discarded. The commit state operation (Eq. 5b) consists merely of saving the last trial state (which is the true equilibrium state) for reference during the next set of state update iterations.
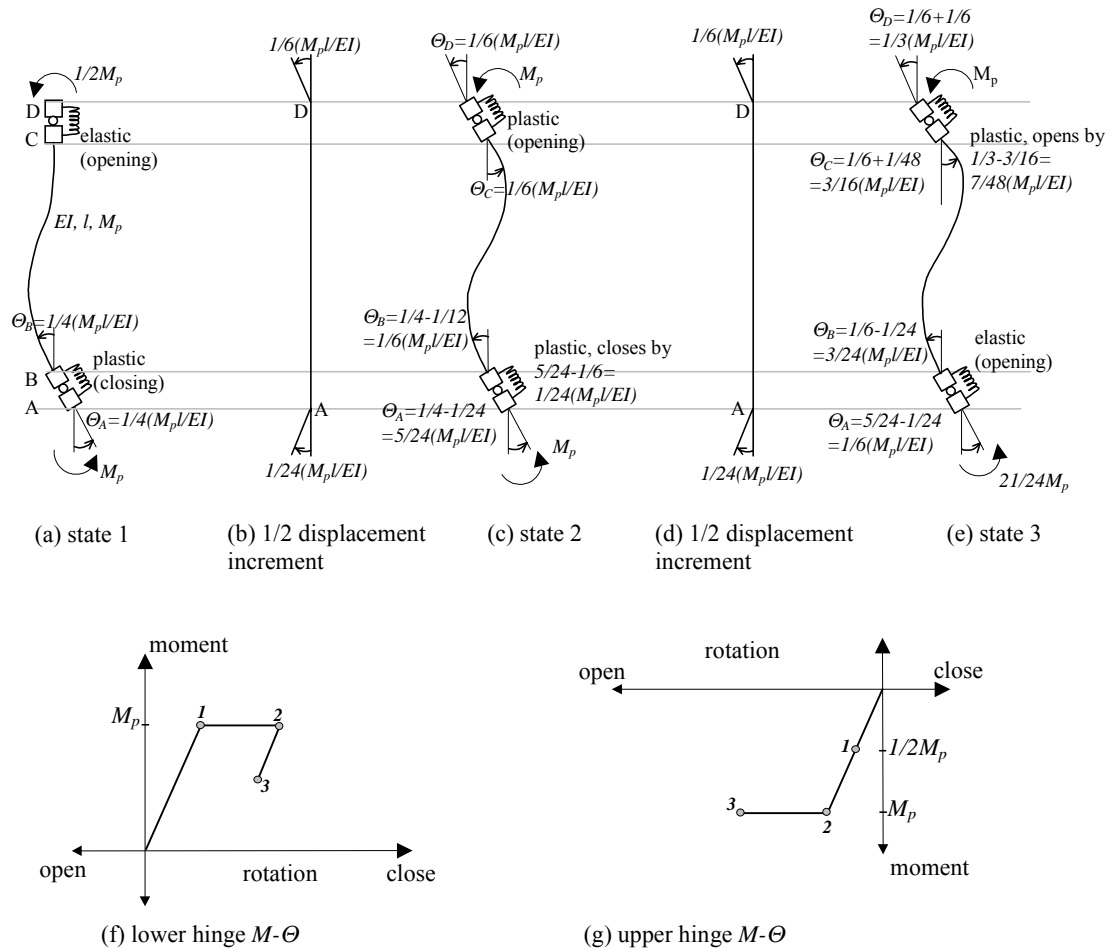

Figure 7. Occurrence of unrecoverable damage during iteration.

7. Application

The proposed algorithm was implemented in a modern finite element program (Archer *et al* [9, 10]) and tested on nine typical steel frame buildings (Krawinkler and Gupta [11]). The sample buildings are located in three different seismic zones (in Los Angeles, Seattle and Boston). For each zone, three different heights of buildings were considered: 3 stories, 9 stories with another one underground, and 20 stories with an additional two underground. The buildings are rectangular in plan, with four to five bays of 6-9 meters in width. The lateral force resisting system consists of exterior ductile moment-resisting frames.

7.1 Analytical model

The models for each of the nine buildings consist of a simple two-dimensional centerline beam-column representation of one exterior moment-resisting frame. Equilibrium was evaluated in the deformed configuration to account for P-Δ effects. The beam-column elements are composed of a central elastic-frame element with nonlinear-inelastic rotational springs at each

end, followed by a rigid end zone. Finite joint dimensions were used in determining the clear span of the members. The nonlinear rotational springs represent the plastic hinges that may develop at the face of the member. Since the algorithm uses an iterative solution technique, a complex curvilinear moment-rotation relationship to account for the spread of plasticity could be used. Instead, in order to test the algorithm, the moment-rotation relationship for the plastic hinge zone is rigid-perfectly-plastic with an AISC/LRFD bilinear interaction curve used to account for the variation in axial force in each member. The bilinear inelastic relationship produces abrupt changes in structural stiffness and the possibility of zero rotational stiffness, both of which are not easily handled by iterative techniques.

7.2 Analysis and static response

FEMA 273 specifies only a monotonic increase in displacement up to the target value. However, to test the stability of the algorithm, the buildings were analyzed for the one complete cycle of displacement (forward and back). The full displacement cycle approximately doubles the negative-stiffness post-peak portion of the load-displacement curve. In addition, the target displacements, which were calculated on the basis of their structural characteristics as well as their location, were increased by 50% to further test the algorithm. The NEHRP design lateral load pattern with an exponent of k=2.0 was used.

One key result of the analysis is the relationship between the base shear and the control node displacement. Figure 8 displays the load-deflection curve for the 20-story building located in Los Angeles. Additional results for the other structures are summarized in Table 1. The curve on Fig. 8 represents the typical nonlinear structural behavior. Up to point A, and in regions C-D and F-G, the structure exhibits elastic behavior. During these stages, the stiffness matrix is positive-definite, and easily handled by the solution technique. In the ranges A-B, D-E and G-H the structural stiffness matrix vanishes. Ranges B-C and E-F are characterized by the unstable equilibrium of the structure, with a negative stiffness matrix. The last two cases clearly represent the two numerical problems stated previously, namely the occurrence of singular and negative stiffness matrix. The presented algorithm successfully overcame these problems.
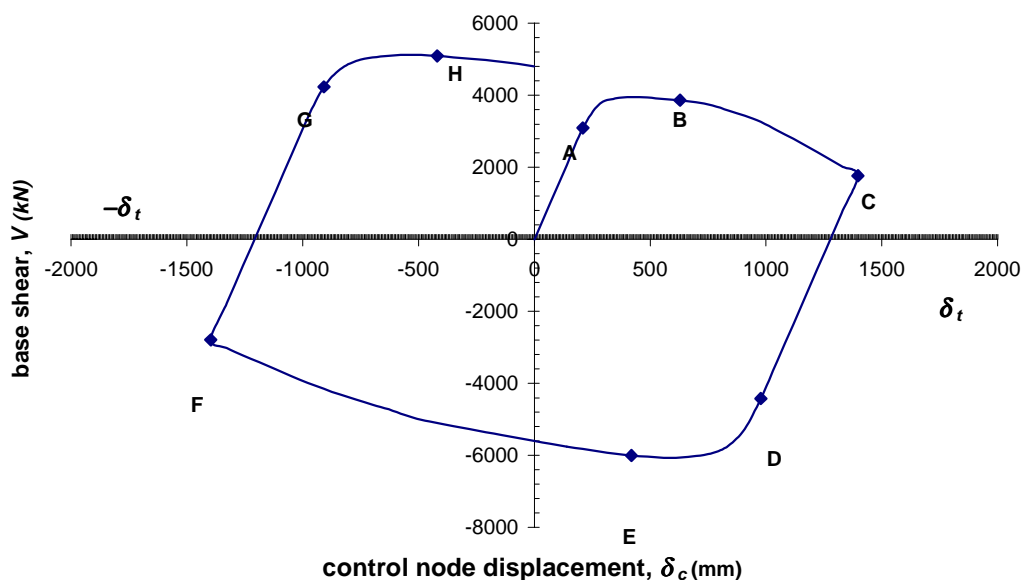

Figure 8. Load-deflection curve for the 20-story building in LA

The solution algorithm is not specific to any particular stiffness-linearization scheme. For the sample buildings, a tangent stiffness matrix (Newton-Raphson iteration) scheme was employed. The number of iterations was small in all cases (see Table 1), and depended on the displacement increment as well as on the prescribed tolerance. In general, minimum number of two iterations occurs at the beginning of the analysis until nonlinear behavior starts to develop (until point A). Two iterations were required, even for the elastic segments of the load-deflection curves, due to the changing axial forces in the columns that in turn affected the geometric stiffness. The maximum number of iterations occurred as structure was displaced up to +/- target displacement (around point C and F). The average number of iterations governed the largest part of the analysis, which covers the descending branches on the load-deflection curves.

Table 1. Summarized results for all of the nine buildings

Location	No. of stories	No. of DOF	No. of NL hinges	δ_τ (mm)	No. of iterations per time step	Max. Base shear (kN)	Max. plastic hinge rotation (rad)
LA	3	51	48	381	aver = 3.0 max = 4	4026	0.0404
	9	198	204	762	aver = 3.1 max = 4	6708	0.0255
	20	354	468	1397	aver = 3.1 max = 5	3951	0.0594
Seattle	3	51	54	254	aver = 2.6 max = 4	3089	0.029
	9	198	194	635	aver = 2.6 max = 4	3984	0.022
	20	354	468	889	aver = 2.2 max = 4	5590	0.013
Boston	3	51	54	152	aver = 2.0 max = 2	1403	0.0112
	9	198	214	254	aver = 2.0 max = 2	2033	0.0065
	20	354	468	381	aver = 2.0 max = 2	3538	0.0042

A control node increment of $\delta_t / 20$ was used as a starting value in the analysis of all the nine buildings. The convergence tolerance for load unbalance was *0.0002 kN mm* for bending moments, and *4.4 N* for shear and axial force. If the required tolerance was not achieved within a dozen iterations, the incremental step was decreased to $\delta_t / 100$. This generally occurred only at extreme levels of displacement where significant damage occurred in the structure, and at displacement reversals that cause abrupt changes in stiffness as the plastic hinges close elastically.

The analyses performed on the nine frames demonstrated the possibility of the local mechanism occurrence. A local mechanism was observed in the analysis of the 3-story building in LA, in which a relatively large target displacement was applied. This caused high deformations in the structure, resulting in the formation of plastic hinges in the first story columns (a first story mechanism was formed). In addition, a local mechanism occurred in the 20-story LA buildings in the lower stories that experienced a large inter-story drift. This resulted in a large plastic deformation at the column ends and caused an inter-story mechanism. Numerically, this means that although the stiffness matrix was not singular for the structure as a whole, it did vanish locally (for one or more stories). This problem was successfully controlled by the use of small fictitious springs attached to the affected nodes.

The presented results demonstrate the capability of the proposed algorithm to successfully deal with a wide range of the nonlinear behavior of building frames, produced by applying the different values of the target displacements to the structure. The results included structural behavior that range from elastic, up to the highly nonlinear behavior with the occurrence of plastic hinges. The proposed algorithm showed to be equally efficient and stable for analyzing the various heights of the building frames.

8. Conclusions

The presented algorithm is based upon requirements of the Nonlinear Static Procedure specified by the NEHRP Guidelines for Seismic Rehabilitation of Buildings. It is formulated for two and three-dimensional analysis, with the two-dimensional analysis initially applied and tested in this work.

The algorithm successfully implements a displacement control technique, which uses a single displacement component as an independent variable, and thus agrees with the nature of the push-over strategy. It involves the use of a single fictitious spring attached to a control node. In addition to being used to produce the desired control node displacement, the spring also properly eliminates the occurrence of global structure instabilities that produce zero and negative linearized stiffness matrices in the analysis. The addition of the spring at the control node does not eliminate the possible formation of local mechanisms in the structure. These local mechanisms are identified in the stiffness matrix decomposition and are eliminated by the addition of a very small, grounded spring to the affected nodes.

Since the structure instabilities are eliminated, the algorithm can employ an iterative solution technique in the search for load equilibrium. Thus, arbitrarily complex curvilinear constitutive relationships can be used. The scheme is not limited to piecewise linear approximations.

<div align="center">NOTATIONS</div>

The following notation is used in this paper:

c	control node,
δ_c	prescribed control node displacement for the current incremental step,
$\Delta\delta_c$	control-node displacement increment,
δ_t	target displacement,
F	the unit lateral load vector,
ΔF	the vector of the lateral load increments,
F_s	the control force,
K	the current stiffness matrix,
K_s	the spring stiffness,
$\lvert K \rvert$	determinant of the tangent stiffness matrix,
$\lVert K \rVert$	norm of the matrix K,
$\lVert K \rVert_\infty$	infinity norm of K,
λ	the lateral load scale factor,
$\Delta\lambda$	the increment of the lateral load scale factor,
Q	the vector of external nodal loads,
$Q\rvert_c$	external load at the control node (includes control force),
Q_G	the vector of gravity loads,
Q_R	the vector of internal resisting loads,
$Q_R\rvert_c$	the resisting force at the control node (includes spring force),
R	the reaction force at the control node
	(R_U - due to unbalanced forces, R_F - due to unit lateral vector),

u the vector of nodal displacements
                (u_U – due to unbalanced forces, u_F – due to unit lateral load),

$u\big|_c$ current control node displacement
                ($u_U\big|_c$ – due to unbalanced forces, $u_F\big|_c$ – due to unit lateral load),

Δu the vector of nodal displacement increments,

U the vector of unbalanced forces,

$\parallel U \parallel$ magnitude of the unbalanced load.

REFERENCES

1. FEMA-273, 1997, "NEHRP Guidelines for the Seismic Rehabilitation of Buildings" Federal Emergency Management Agency Publication 273.

2. FEMA-274, 1997, "NEHRP Commentary on the Guidelines for the Seismic Rehabilitation of Buildings" Federal Emergency Management Agency Publication 274.

3. Argyris, J. H., 1965, Continua and Discontinua, Proceedings of the first Conference in the Matrix Methods in Structural Mechanics, Section V – Rehearsal of Finite Displacement Theory and Applications, Wright-Patterson A.F.B., Ohio.

4. Pian, T. H., and Tong P., 1970, Variational Formulation of Finite-Displacement Analysis, Symposium on High Speed Computing of Elastic Structures, University of Liege, 43 - 63.

5. Zienkiewicz, O. C., 1971, Incremental Displacement in Non-Linear Analysis, International Journal for Numerical Methods in Engineering, 3, 587 - 588.

6. Sharifi, P., and Popov, E. P., 1971, Nonlinear Buckling Analysis of Sandwich Arches, Journal of the Engineering Mechanics Division, 1397 - 1412.

7. Batoz, J-L., and Dhatt, G., 1979, Incremental Displacement Algorithms for Nonlinear Problems, International Journal for Numerical Methods in Engineering, 14, No 1, 1262 - 1267.

8. Clarke, M.J., and Hancock, G.J., 1990, A Study of Incremental-iterative Strategies for Non-linear Analyses, International Journal for Numerical Methods in Engineering, 29, 1365-1391

9. Archer, G. C., 1996, Object-Oriented Finite Element Analysis, Ph.D. Dissertation, University of California at Berkeley.

10. Archer, G. C., Fenves, G., and Thewalt, C., 1999, A new object-oriented finite element analysis program architecture, Computers and Structures, 63-75.

11. Krawinkler, H., and Gupta, A., 1998, "Story drift demands for steel moment frame structures in different seismic regions". Proceedings, Sixth U.S. National Conference on Earthquake Engineering, Earthquake Engineering Research Inst., Oakland, California.