ASIAN JOURNAL OF CIVIL ENGINEERING (BUILDING AND HOUSING) VOL. 5, NOS 3-4 (2004)
PAGES 145-160

PLASTIC ANALYSIS OF PLANAR FRAMES USING KINEMATIC METHOD AND GENETIC ALGORITHM

A. Kaveh* and M. Jahanshahi
Department of Civil Engineering
Iran University of Science and Technology, Narmak, Tehran-16, Iran

ABSTRACT

Plastic analysis using kinematic approach is quite conventional. A C++ program has been developed which computes the collapse load factor of planar frames using combination of elementary mechanisms and Genetic algorithms. The program has the capability of generating the frames visually and observing the results by simple clicks on appropriate icons. Elementary and combined collapse mechanisms could be viewed on screen instead of having to interpret the results from the rush of outputs. Many examples presented here show how easy it is to create the model from scratch, perform the loading, and complete the analysis.

1. INTRODUCTION

The minimum and maximum principles form the basis of all the general analytical methods for plastic analysis and design, Baker, Horne and Heyman [1]. The most widely applicable method of analysis based on the minimum principle is that of the combination of elementary mechanisms, developed by Neal and Symonds [2-4].

Plastic analysis and design of rigid-jointed frames has been cast in the form of linear programming by Charnes and Greenberg [5], as early as 1951. Further progress in the field is due to Heyman [6], Horne [7], Baker and Heyman [8], Jennings [9], Watwood [10], Gorman [11], and Theirauf [12], and Kaveh [13], among many others. Considerable progress has been made in the past decade: a complete list of which may be found in Munro [14] and Livesley [15]. Plastic analysis using combination of elementary mechanisms could be used as a method to find the collapse load factor of planar frames, but this method of analysis has some drawbacks which limit its application. For instance, as the structure becomes more complex, the computation of elementary mechanisms turns out to be hard and tedious. Also, if the collapse mechanism deemed for the structure and its assumed loading is not the correct one, the resulting collapse load will be an upper bound to the actual collapse load. With a microcomputer and a suitable program at hand, the problems mentioned above could be solved. In this work, an efficient program is developed for plastic analysis of planar

* E-mail address of the corresponding author: alikaveh@iust.ac.ir

frames. This is not only a simple analysis program, but a program to ease the work of designer in creating his models and trying as many alternatives as he wishes. Basic and combined mechanisms generated by the program in the combination phase could be viewed. This is of great importance especially when the designer would like to see the mechanisms with the same load factors. Other aspects of the present program include plotting and full editing capabilities.

2. GENERATION OF ELEMENTARY MECHANISMS

In order to find a set of independent mechanisms, the method of Watwood [11] could be used. In this method joint mechanisms are also computed which is unnecessary because joint mechanisms could automatically be assigned to each joint. Axial deformations could also be neglected. With the elimination of these two, we are led to the method used by Pellegrino and Calladine [16] and Deeks [17].

In this method which finds the independent mechanisms for an assembly of pinned joint rigid bars, two degrees of freedom are assigned to each unrestraint joint. The elongation of each member is expressed in terms of the displacements of end joints of that member in global coordinates.

$$e = (d_{xk} - d_{xi}) \cos\alpha + (d_{yk} - d_{yi}) \sin\alpha \tag{1}$$

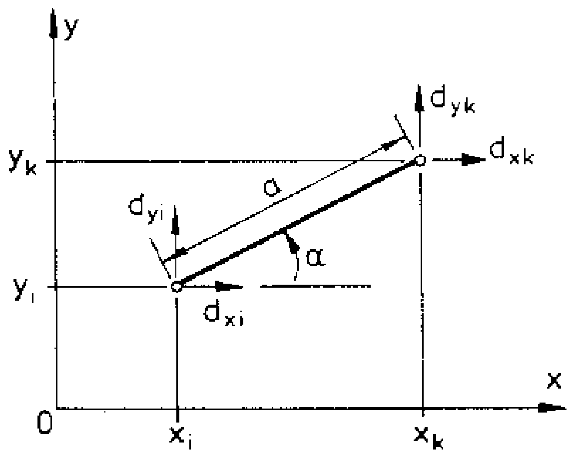


Figure 1. Degrees of freedom of a typical member

By writing all such expressions for all members, the resulting equation will be as follows:

$$\mathbf{e} = \mathbf{Cd}. \tag{2}$$

In a valid mechanism, members do not elongate, therefore to find basic mechanisms, it is necessary to solve the following equations:

$$\mathbf{Cd} = \mathbf{0}. \tag{3}$$

Since the frame is not a truss and since it is not stable with all the joints pinned, number of columns of **C** exceeds the number of rows and the difference is the number of independent mechanisms. By performing Gaussian elimination on Eq. (3), it is reduced to the following:

$$[\mathbf{I} \mid \mathbf{C}_\mathrm{d}] \begin{Bmatrix} \mathbf{d}^\mathrm{d} \\ \mathbf{d}^\mathrm{i} \end{Bmatrix} = 0 \tag{4}$$

In other words, columns relating to dependent displacements $\mathbf{d}^\mathrm{d}$ are reduced to identity matrix **I**. By rearranging this equation, $\mathbf{d}^\mathrm{d}$ could be expressed in terms of $\mathbf{d}^\mathrm{i}$ as:

$$\mathbf{d}^\mathrm{d} = -\mathbf{C}_\mathrm{d}\mathbf{d}^\mathrm{i} \tag{5}$$

Choosing independent vectors for $\mathbf{d}^\mathrm{i}$ (equal to the number of independent mechanisms) and computing $\mathbf{d}^\mathrm{d}$ thereafter, the solution to Eq. (3) results in independent mechanisms. As a simple computational approach, the independent vectors could be formed each time by setting one of the independent displacements to unity and the others to zero.

Mechanisms obtained in this way may be used in linear programming methods, but are not suitable for combination of mechanisms. They contain more active hinges than is necessary for a logical collapse mechanism. A collapse mechanism should become determinate by removing just one active hinge. An independent mechanism computed by this method may contain all active hinges of another mechanism.

Following the method of Deeks [17], independent mechanisms could be purified by removing the excess hinges to represent a set of potential collapse mechanisms. In this method, every independent mechanism is checked to see whether it contains any other mechanism. If so, it is purified by removing the contained mechanism. This process is repeated until no more modifications can be made.

3. DETERMINATION OF COLLAPSE LOAD

Collapse load factor is obtained by the virtual work theorem. Rotations and displacements are considered to be virtual and the internal and external virtual works are computed. The collapse load factor for a specific mechanism is the ratio of these two.

$$\lambda_\mathrm{c} = \frac{\text{int ernal virtual work}}{\text{external virtual work}} \tag{6}$$

The external virtual work is calculated by multiplying the joint forces **P**, by the joint displacements **d**.

$$\text{external virtual work} = \mathbf{P}^t\mathbf{d}. \tag{7}$$

The internal virtual work is calculated by multiplying the rotations at the plastic hinges (**r**) by the plastic moments of members in which the plastic hinges form ($\mathbf{M}_p$). Since the plastic moments always resist the rotations at hinges, therefore absolute values are used.

$$\text{internal virtual work} = \mathbf{M}^t|\mathbf{r}|. \tag{8}$$

Since the joint mechanisms have been neglected during the formation of independent mechanisms, it is necessary to find the location of hinges in members. This location is so determined to minimize the internal virtual work.

If a joint is restraint against rotation, hinges are formed in all members connecting to the joint. However, if the joint is not restraint, hinges are formed in n-1 members among the n members connected to that joint. In this case, n possible locations for hinges can exist and all of them should be checked for minimum internal work. When less than the maximum number of hinges form, the rotation in one or more of the assumed hinges is zero and does not contribute to the virtual work.

4. COMBINATION OF ELEMENTARY MECHANISMS

After generating elementary mechanisms, the next step is to combine suitable mechanisms in order to obtain a logical collapse mechanism with lowest load factor. An experienced analyzer would choose the correct mechanisms in a short time, but in a computer oriented method this does not work. All possible combinations should be taken into account. Some mechanisms may increase the load factor in combination with the current one, but they may reduce it after combination with other mechanisms. Therefor when a mechanism does not reduce the load factor, this does not mean that it should be excluded. The criteria for a collapse mechanism to be a correct one is that there is no possibility in reducing the load factor in combination with any other mechanism.

Using the method of Neal and Symonds [12], an algorithm can be designed to check every possible combination by starting with an elementary mechanism and combining that mechanism with others in order to reduce the load factor. When no more reduction can be achieved by combining elementary mechanisms with the current one, the same process is repeated with the next elementary mechanism until no more possibility is left. Thus the collapse load factor is the lowest load factor obtained in this process.

Some authors use recursive algorithms to combine mechanisms, but recursive algorithms have the deficiency that they may shut down even for moderate frames. A better implementation would be to use an endless loop with a good condition for terminating the loop when the possibilities are exhausted. The following code fragments show how to do this:

```
void
```

```
TFailure :: CombineMechanisms ()
{
  mechIncluded = new bool [numIndepMechs];

  for (unsigned i = 0; i < numIndepMechs; i ++)    {
    for (unsigned j = 0; j < numIndepMechs; j ++)
      mechIncluded [j] = false;

    TMechanism tempMech (IndepMechs [i]);
    mechIncluded [i] = true;

    SaveCombMech (tempMech);

    mechsProcessed ++;

    bool done;
    do    {
      done = true;

      for (unsigned j = 0; j < numIndepMechs; j ++)
        if (!mechIncluded [j])    {
          double collapseLoadFactor = tempMech . GetCollapseLoadFactor ();

          Combine (tempMech, IndepMechs [j]);

          if (collapseLoadFactor - tempMech . GetCollapseLoadFactor () >
PRECISION)    {
            mechIncluded [j] = true;

            SaveCombMech (tempMech);

            done = false;
          }

          mechsProcessed ++;
        }
    } while (!done);
  }

  delete[] mechIncluded;
}
```

The main loop in the *CombineMechanisms* () procedure terminates only when no more mechanisms can be combined with the current one in the inner loop. This is identified by the status of the flag *done*. Whenever a mechanism is combined with the current one, *done* takes the value of *false*; otherwise it keeps its original value which has previously been set to *true*. If the value of *done* is not changed throughout the inner loop, meaning that no more mechanism can be combined, the main loop terminates. Two points which should be mentioned here are that no more mechanism should be combined twice and this is taken care of by *mechIncluded* array which keeps track of the mechanisms being combined and second two mechanisms are only combined when their combination leads to a reduced load factor.

5. GENETIC ALGORITHMS

In this section a second approach based on Genetic algorithm is presented for calculation of the collapse load factor of planar frames. These algorithms are used to choose appropriate elementary mechanisms to be combined and find the lowest load factor. For this purpose, some preliminary definitions are necessary.

Chromosomes are strings of binary bits, in this case they consist of zeros and ones equal to the number of independent mechanisms. A one for a bit means that the corresponding mechanism will take part in combination otherwise it does not.

1 2 3 4 5 6 7 8	Mechanism:
$\boxed{1\ 1\ 0\ 1\ 1\ 0\ 1\ 1}$	Chromosome:

In the previous example, mechanisms 1, 2, 4, 5, 7, 8 are combined and mechanisms 3 and 6 do not.

Crossover is an operation in which two strings are crossed and new strings are generated. For the two strings elected for this purpose, a crossing site is selected with the uniform probability P_c between the first and last bits. The bits extending from the crossing site to the end of the strings are exchanged. The following example illustrates the idea.

$$A_1 = 0\ 1\ 1\ 0\ 1$$
$$A_2 = 1\ 1\ 0\ 0\ 0$$

A crossing site has been selected between bits 4 and 5. After crossover, the following strings are generated

$$A'_1 = 0\ 1\ 1\ 0\ 0$$
$$A'_2 = 1\ 1\ 0\ 0\ 1$$

Mutation is the random change of a randomly selected bit from 1 to 0 or vice versa.

To begin the search for the lowest load factor, an initial generation is produced randomly and Genetic operations are performed on it. Fittest individuals are copied to a new generation and the same process is repeated until a fair approximation is reached. The measure for fitness is the fitness function which is defined as follows:

$$f_i = C - \lambda_i$$

In which f_i, $\boldsymbol{\lambda}_i$ and C are the fitness function for chromosome i, the corresponding load factor and the maximum load factor in the current generation, respectively. Thus the problem of minimizing the load factor is transformed to maximization of the fitness function.

6. NUMERICAL RESULTS

Example 1: A gable frame is considered as shown in Figure 2. The configuration and loading are shown in this figure. The frame has four independent mechanisms which are shown in Figure 3. The collapse load factor from the direct method is obtained as 0.8182 and the collapse mechanism is shown in Figure 4. The results are in good agreements with those of Deeks [17], who has solved the problem using the same method.

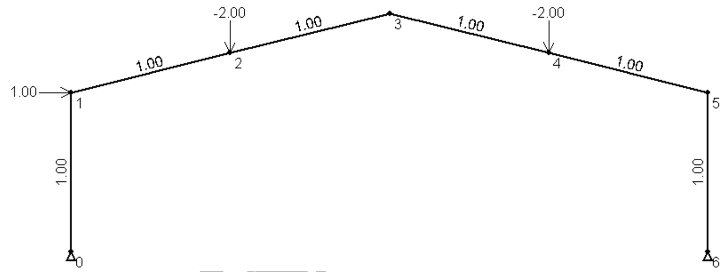
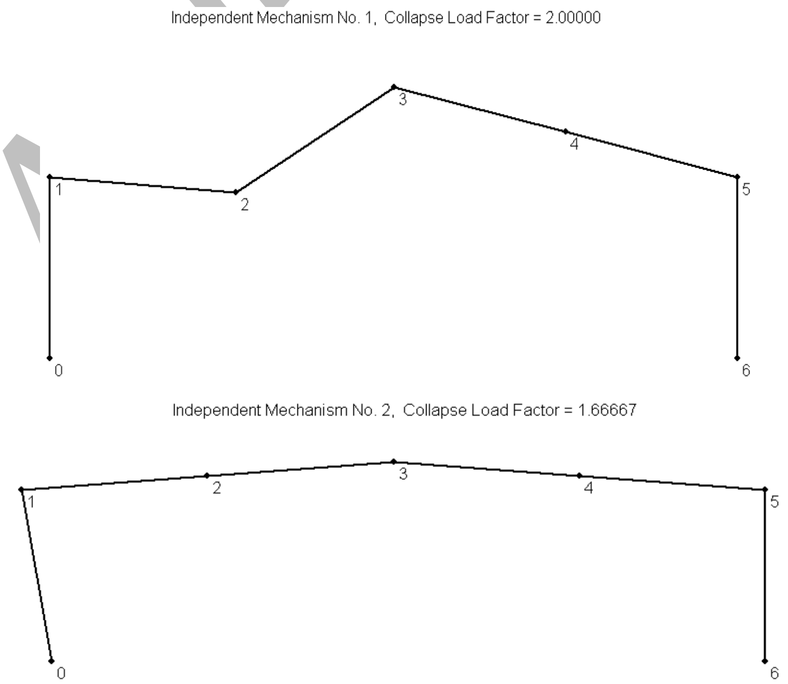


Figure 2. A gable frame, loading and plastic moments of members



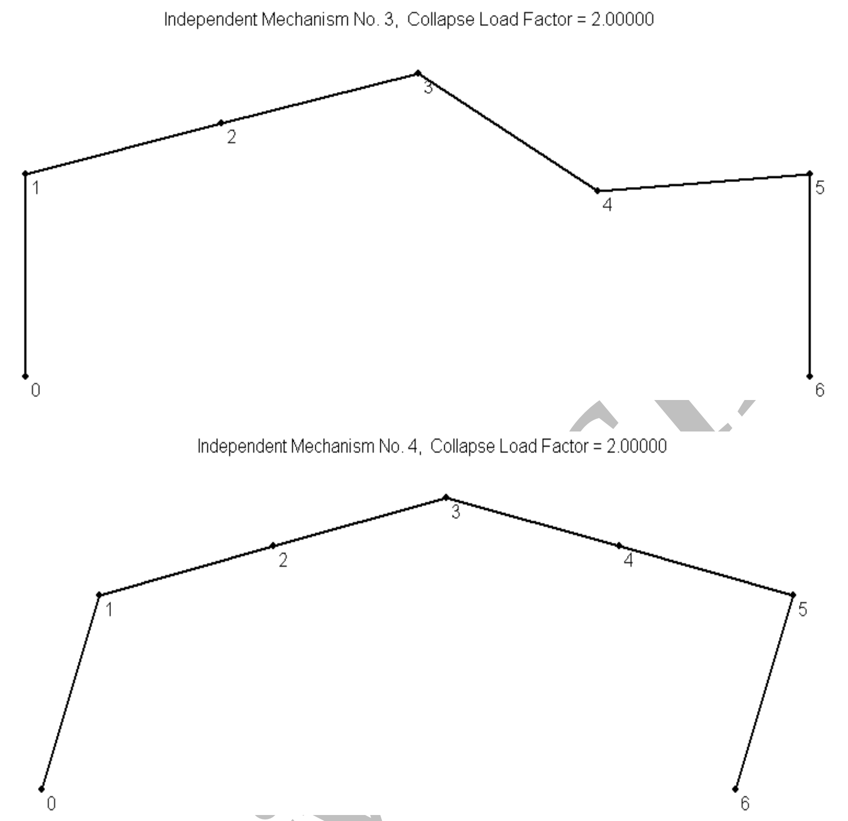


Figure 3. Elementary mechanisms

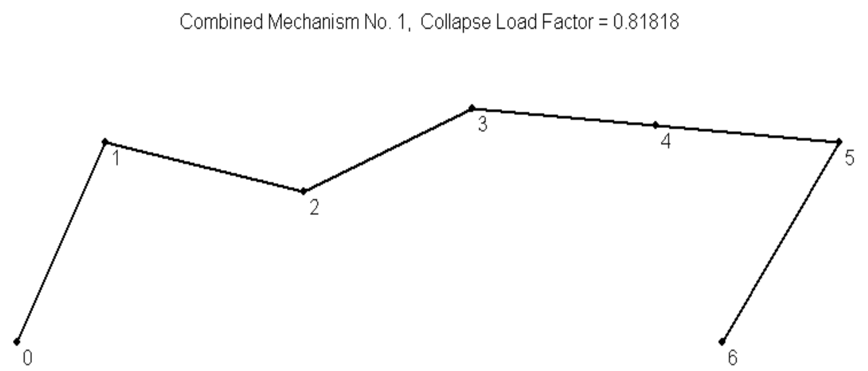


Figure 4. Collapse mechanism

Example 2: This example is a three-story frame. The actual collapse load factor calculated by the direct method is 1.97 and that by Genetic algorithm results in 2.00 However, the collapse mechanism from Genetic algorithm is quite different.

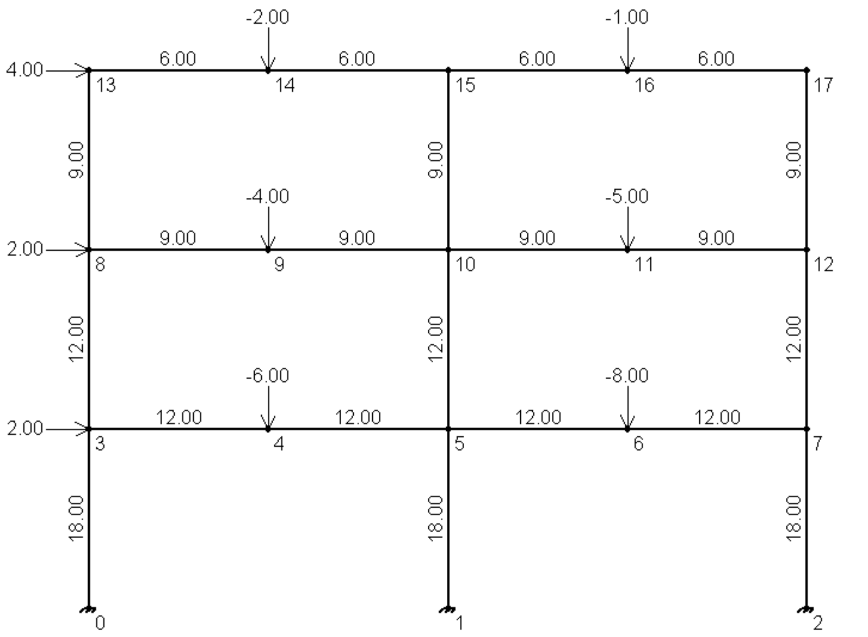
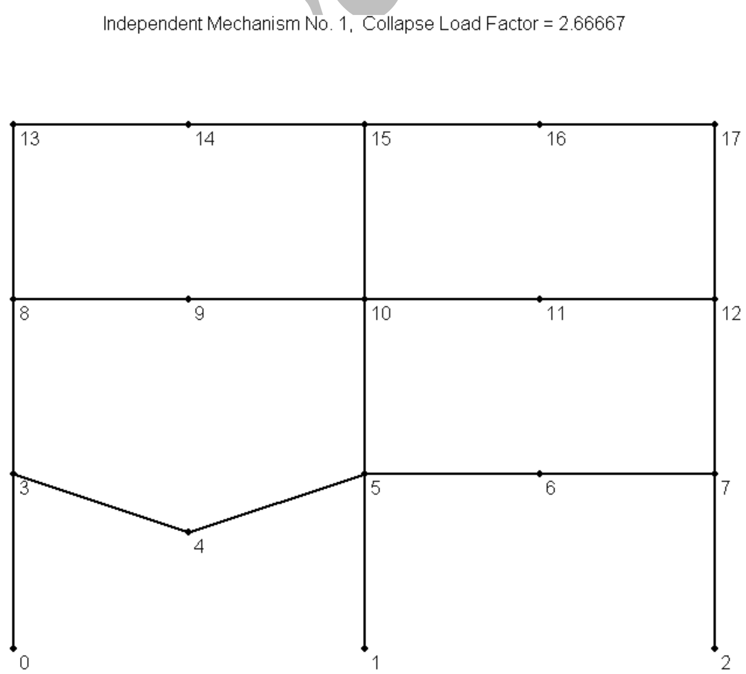


Figure 5. A three-story frame, loading and plastic moments of members



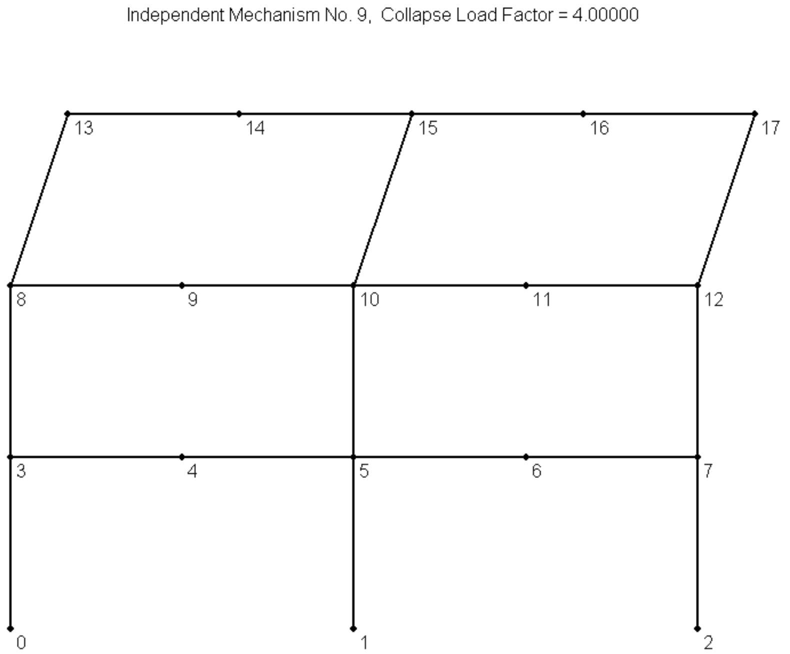
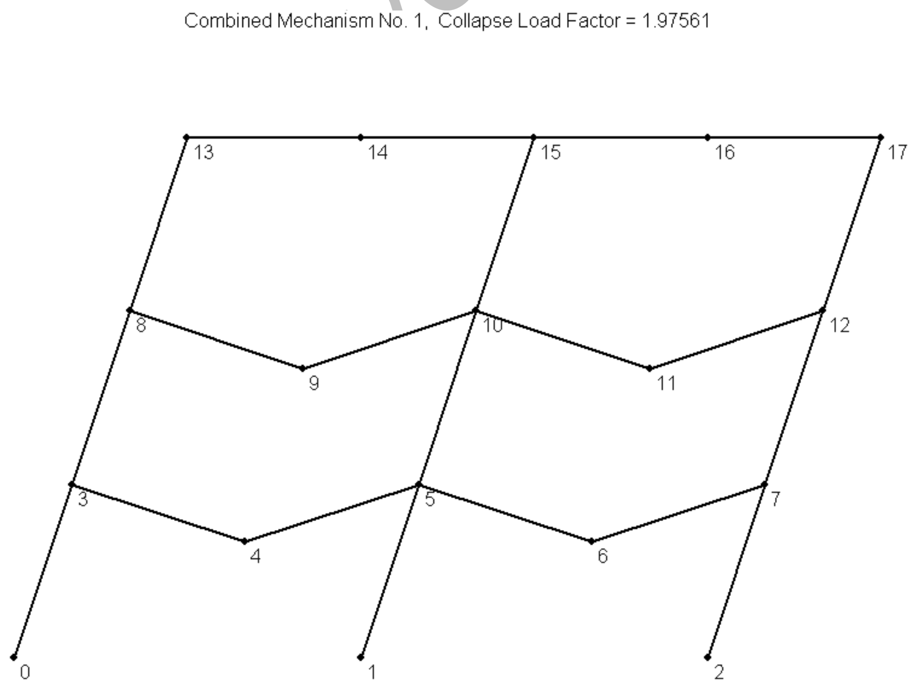


Figure 6. Independent mechanisms 2 and 9



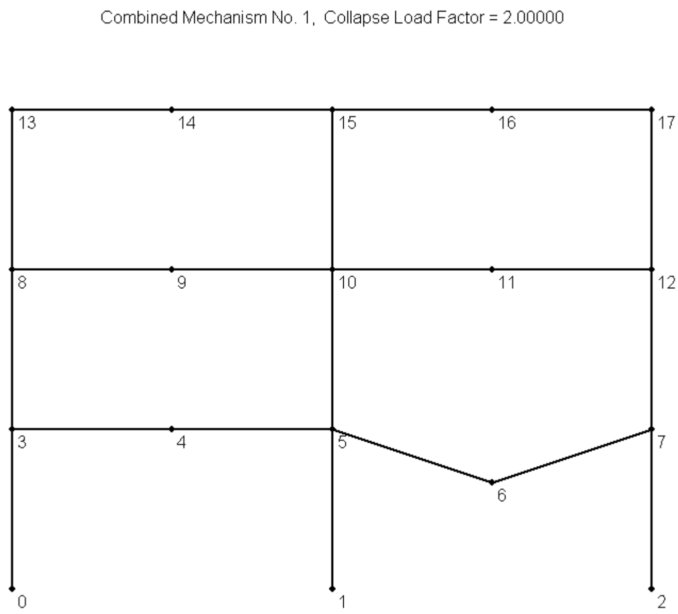


Figure 7. Collapse mechanisms obtained by direct method and Genetic algorithms

Example 3: A six-story frame is considered as shown in Figure 8. The load factors and collapse mechanisms obtained by direct method and Genetic algorithm are both the same. Three of the elementary mechanisms are shown in Figure 9 and the collapse mechanism is illustrated in Figure 10.

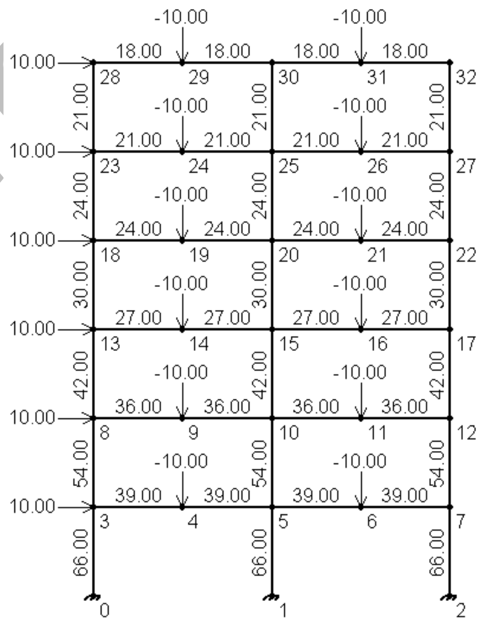
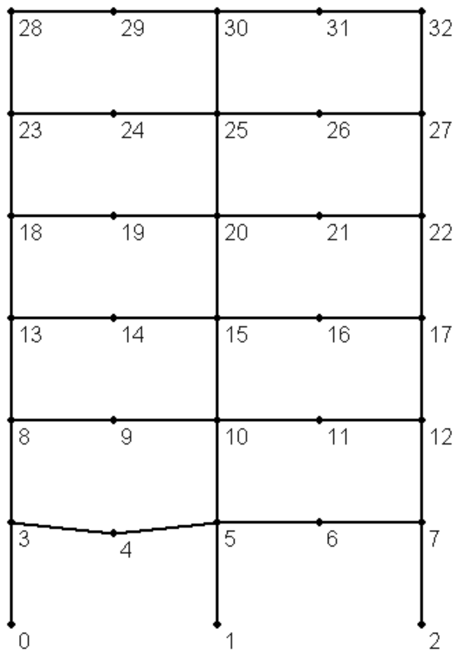
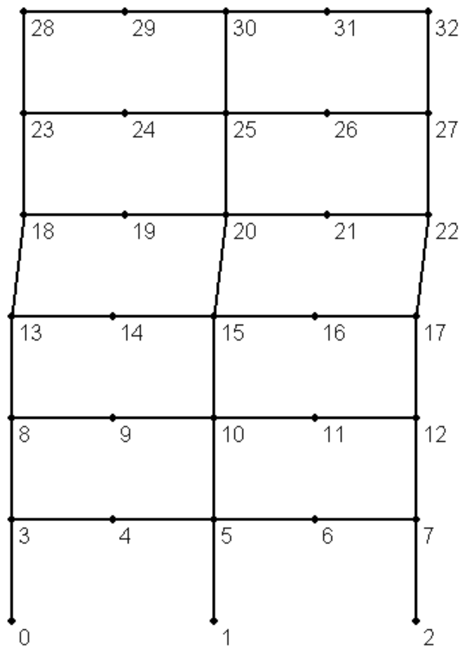


Figure 8. A six-story frame, loading and plastic moments of members

Independent Mechanism No. 1, Collapse Load Factor = 5.20000



Independent Mechanism No. 12, Collapse Load Factor = 2.00000



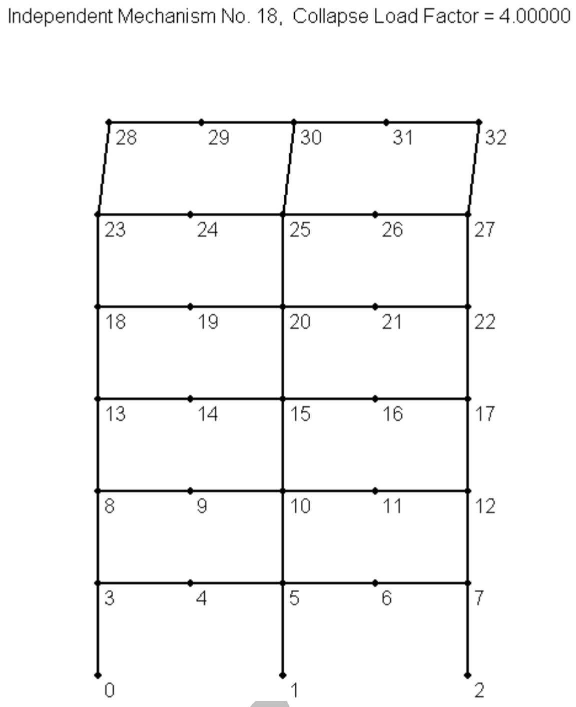


Figure 9. The first, twelve and eighteenth elementary mechanisms

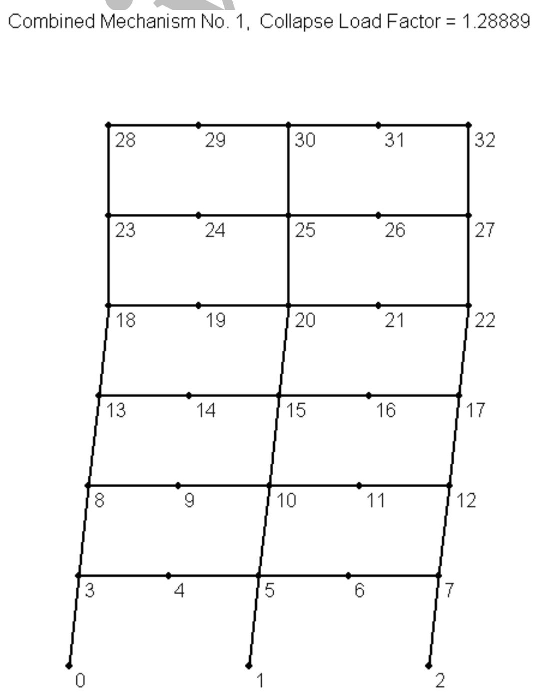


Figure 10. The collapse mechanism

Example 4: A four-story frame is considered as shown in Figure 11. This is a good example for showing the efficiency of the Genetic algorithms when applied to general problems. The correct collapse mechanism and the mechanism obtained by Genetic algorithm are shown in Figure 12.

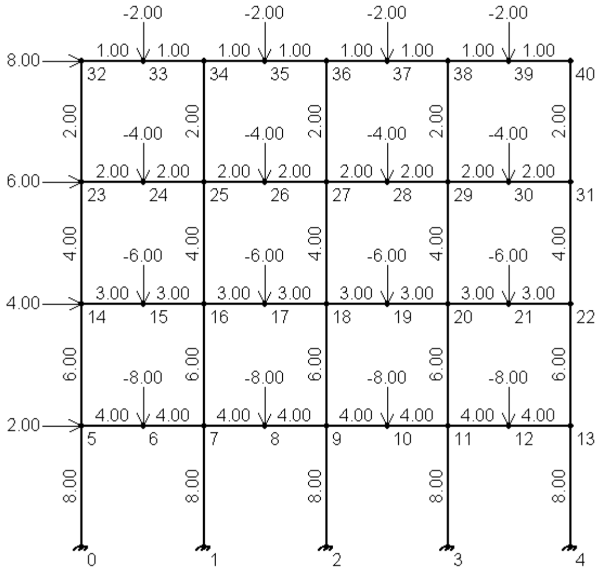
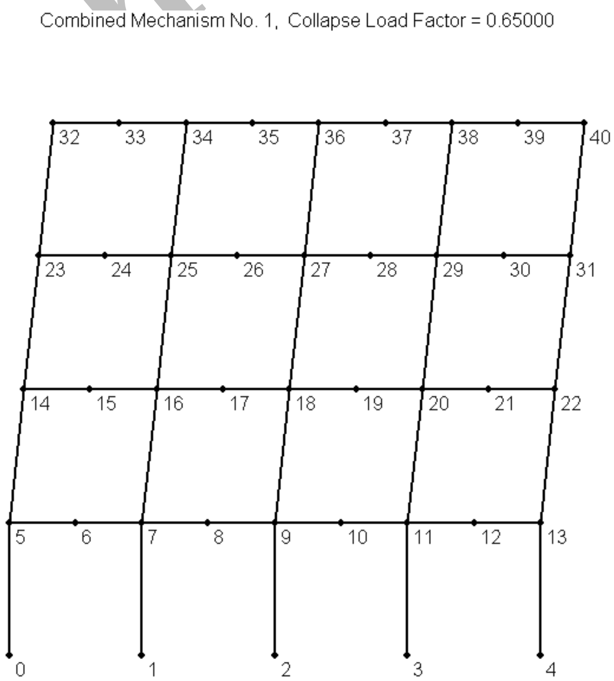


Figure 11. A four-story frame, loading and plastic moments of members



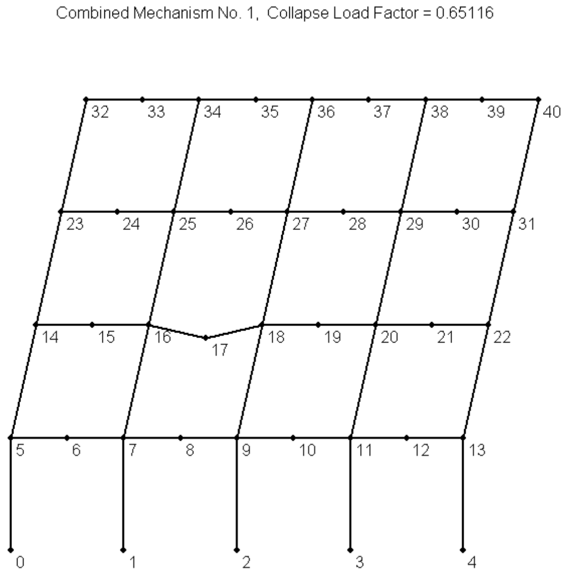


Figure 12. The correct mechanism and the mechanism obtained by Genetic algorithm

7. CONCLUDING REMARKS

In Table I, a comparison has been made between the actual load factors of the previous examples and those obtained by Genetic algorithm.

It is obvious that Genetic algorithm gives good approximations to the actual load factors, but if the correct collapse mechanism is deemed, this algorithm should be used cautiously.

Table 1: Actual load factors and those obtained by Genetic algorithm

Example	**Actual Load Factor**	**Load Factor by Genetic Algorithm**	**Difference %**
2	1.97561	2.00000	1.23
3	1.28889	1.28889	0.00
4	0.65000	0.65116	0.18

REFERENCES

1. Baker J., Horne M.R, and Heyman J., *The Steel Skeleton; Plastic Behaviour and Design*, Cambridge at the University Press, Volume 2, 1956.

2. Neal, B.G. and Symonds P.S., The rapid calculation of plastic collapse loads for a framed structures, Proc. Instits Civ. Engrs, Part 3, Volume 1, 1952.
3. Neal, B.G. and Symonds P.S., The calculation of plastic loads for plane frames, Preliminary Publication, International Association for Bridge and Structural Engineering, 4th Congress, Cambridge and London, 1952.
4. Neal, B.G. and Symonds P.S., The calculation of collapse loads for framed structures, *J. Instits Civ. Engrs*, Volume 35, 1950-1.
5. Charnes A. and Greeberg H.J., Plastic collapse and linear programming, Summer Meeting of the American Mathematical Society, 1959.
6. Heyman, J. On the minimum weight design of a simple portal frame, *International Journal of Mechanical Science*, **1**(1960)121-134.
7. Horne M.R. Determination of the shape of fixed ended beams for maximum economy according to the plastic theory, Final Report, International Association of Bridge and Structural Engineering, 4th Congress: Cambridge London, 1953.
8. Baker J., Heyman, J. *Plastic Design of Frames, Fundamentals*, Vol. 1, Cambridge, Cambridge University Press, 1969.
9. Jennings, A. Adapting the simplex method to plastic design, In: Marris, LJ, editor. Proceedings of Instability and Plastic Collapse of Steel Structures, 1983, pp. 164-173.
10. Watwood, V.B. Mecahanism generation for limit analysis of frames, *J. Struct. Div., ASCE*, ST1, **109**(1979)1-15.
11. Gorman, M.R., Automatic generation for limit analysis of frames, *Proc. ASCE*, ST7, (1981)1350-1354.
12. Thierauf, G. A method for optimal limit design of structures with alternative loads, Computational Methods in Applied Mechanics in Engineering, **16**(1987)134-149.
13. Kaveh, A. *Optimal Structural Analysis*. RSP (John Wiley), UK, 1997.
14. Munro, J. Optimal plastic design of frames In Proceedings of NATO, Waterloo: *Advanced Study in Engineering Plasticity by Mathematical Programming*, 1977.
15. Livesley, R.K. Linear programming in structural analysis and design. In: Gallagher RH et al editors, Optimum Structural Design, Chapter 6, New York, Wiley, 1977.
16. Pellegrino S. and Calladine, C.R. Structural computation of an assembly of rigid links, frictionless joints, and elastic springs. *J. Appl. Mech. ASME*, **58**(1991)749-753.
17. Deeks, A.J. Automatic computation of plastic collapse loads for frames, *Computers and Structures*, No. 3, **60**(1996) 391-102.