Gray Coding in Evolutionary Multicriteria Optimization: Application in Frame Structural Optimum Design

David Greiner, Gabriel Winter, José M. Emperador and Blas Galván

Evolutionary Computation and Applications Division (CEANI)
Intelligent Systems and Numerical Applications in Engineering Institute (IUSIANI)
35017, Campus de Tafira Baja, Las Palmas de Gran Canaria University (ULPGC), Spain
dgreiner@iusiani.ulpgc.es

Abstract. A comparative study of the use of Gray coding in multicriteria evolutionary optimisation is performed using the SPEA2 and NSGAII algorithms and applied to a frame structural optimisation problem. A double minimization is handled: constrained mass and number of different cross-section types. Influence of various mutation rates is considered. The comparative statistical results of the test case cover a convergence study during evolution by means of certain metrics that measure front amplitude and distance to the optimal front. Results in a 55 bar-sized frame test case show that the use of the Standard Binary Reflected Gray code compared versus Binary code allows to obtain fast and more accurate solutions, more coverage of non-dominated fronts; both with improved robustness in frame structural multiobjective optimum design.

1 Introduction

Recently, interest in analysis and design of representations and operators for evolutionary computation has been liven up (e.g. a special Issue about this topic of the journal IEEE Transactions on Evolutionary Computation is coming). The motivation of this work is to analyse the influence of an adequate coding in multicriteria optimization, particularly we compare here the use of Gray coding versus binary coding. In multiobjective optimization the search has to deal with multiple requirements: the approximation to the optimum non-dominated front, the achievement of a smooth distribution along the front and also the completion of its maximum coverage [6][8]. So, the codification influence in the search towards the set of optimum solutions should be focused in such a plural way. The choice of the proper coding can have a drastic repercussion in the final results. The smoothness in the correspondence between the phenotypic and the genotypic space is the main claimed advantage of the Gray Code [29][30]. Guarantying this smoothness could be especially critical when the genotypic unit (represented with 0s and 1s) has its phenotypic correspondence in an ordered database, where each gene has a set of associated values, whose magnitudes can vary considerably even in consecutive genes. This is a frequent case when using discrete representation of the chromosome via 0s and 1s, for example in scheduling optimisation problems [10].

Here a discrete frame structural multicriteria optimization problem belonging to that archetype is handled. The first application of evolutionary algorithms to structural optimization is dated twenty-five years ago [13]: A ten-bar truss is optimized for the minimum constrained mass problem, with continuous variables for the section area of the bars. A pioneer article for frame structures optimization using evolutionary algorithms is [18], where a genetic algorithm is used for the optimal design of skeletal building structures considering discrete sizing, geometrical and topological variables in two design examples. A recent state of the art of structural optimization with special emphasis in evolutionary optimization is [1], where the recent developments in the field, for the period 1980 to 2000, are documented by the ASCE (American Society of Civil Engineering) Technical Committee on Optimal Structural Design of the Technical Administrative Committee on Analysis and Computation. Interesting reviews about multicriteria optimization in structural engineering are [4][5][23], and a set of applications of multicriteria evolutionary optimization in structural and civil engineering are summarized in [6].

The organization of this paper is described as follows: First, the multiobjective frame structural problem is described. Section 3 disserts about the evolutionary approach and the use of Gray Code in multiobjective frame optimization. Section 4 exposes the 55 bar-sized test case. After that, the experimental results are shown in section 5, ending with the conclusions section.

2 Frame Structural Optimum Design

2.1 Definition of the Problem

The frame structural design problem considered has two conflicting objectives: the minimization of the constrained mass and the minimization of the number of different cross-section types considered in the final design. This design problem was introduced in [11], being solved using a combination of weights. It has been solved with elitist multiobjective evolutionary algorithms in [14][15]. Both objectives are explained as follows.

The first objective, the *minimization of the constrained mass* is taken into account to minimize the raw material cost of the designed structure. The constraints consider those conditions that allow the designed frame to carry out its task without collapsing or deforming excessively. The constraints are the following, taking into account the Spanish design code (EA-95) guidelines:

a) Stresses of the bars: where the limit stress depends on the frame material and the comparing stress takes into account the axial and shearing stresses by means of the shear effort, and also the bending effort (a common value for steel is of 260 MPa - S275JR steel -), for each bar:

$$\sigma_{co} - \sigma_{\lim} \leq 0 \quad (1)$$

b) Compressive slenderness limit: where the $\lambda_{\lim}$ value is 200 (to include the buckling effect the evaluation of the β factor, is based on Julian and Lawrence criteria). For each bar:

$$\lambda - \lambda_{\lim} \leq 0 \tag{2}$$

c) Displacements of joints (in each of the three possible degrees of freedom) or middle points of bars. In the test cases, the maximum vertical displacement of each beam is limited (in the multiobjective test case the maximum vertical displacement of the beams is L / 500):

$$u_{co} - u_{\lim} \leq 0 \tag{3}$$

The first objective function *constrained mass*, results:

$$ObjetiveFunction_1 = \left[\sum_{i=1}^{Nbars} A_i \cdot \rho_i \cdot l_i \right] [1 + k \cdot \sum_{j=1}^{Nviols} (viol_j - 1)] \tag{4}$$

where :

A_i = bar i cross-section area; ρ_i = bar i specific mass; l_i = length of bar i; k = constant that regulates the equivalence between mass and constraints; $viol_j$ = for each of the violated constraints (stress, displacement or slenderness), is the quotient between the value that violates the constraint limit: violated constraint value and its reference limit. The constraints reference limits are chosen according to the Spanish design codes. So, constraints if violated, are integrated into the mass of the whole structure as a penalty depending on the amount of the violation (for each constraint violation the total mass is incremented):

$$viol_j = \frac{Violated\ Constraint\ Value}{Constraint\ Limit} \tag{5}$$

Moreover, the minimization of a second function is considered as a multicriteria optimization problem: the *number of different Cross-Section Types* (*CST*), that supposes a condition of constructive order, and with special relevancy in structures with high number of bars [11][19]. It helps to a better quality control during the execution of the building site. It is a factor that has been also recently related with the life cost cycle optimization of steel structures, as claimed in [24][28], and evolutionary multicriteria optimization allows it to be integrated in the design optimization process.

2.2 The System Model

Numerous literature about evolutionary bar optimization problems is about trusses. They are characterized for its articulated nodes between bars, where no resistant moment is executed and the only required geometric magnitude of the bar section is its area. Nevertheless, here we handle with frames: the nodes between bars are rigid, and the moments have to be taken into account. More bar geometric magnitudes have to be considered: to evaluate the normal stresses, the area, the modulus of section and the relation of beam height are required; to evaluate the shearing stresses the web area is required; to evaluate the medium span displacement the moment of inertia is required. Moreover, considering the buckling effect implies to take into account the radius of gyration. Because of the design is performed using real cross-section types -developing a discrete optimization problem with direct real application-, all these magnitudes are stored in a vector associated to each cross-section type, whose complete set constitute a database. The codification of the chromosome implies for each bar of the structure, a discrete value that is assigned to the order of cross-section types in the database.

The structural calculation implies the resolution of a finite element modelling -with Hermite approximation functions-, and its associated linear equation system. In the plane case, each node has three degrees of freedom: horizontal, vertical and gyration displacements (U), as well as three associated force (F) types: horizontal and vertical forces and momentum. Forces and displacements are related (F=KU) by the stiffness matrix (K), which depends on geometric and material properties, and is created with an assembling process from each element of the structure. A renumbering using the inverse Cuthill-McKee method [7] is used to reduce the matrix bandwidth, in order to reduce the calculation time of the system (essential when many evaluations are required), which is programmed in C++ language.

3 The Evolutionary Approach

The frame structural problem of determining the constrained minimum mass has many local minima [21], so a global optimization method is recommended. Moreover, we deal with a discrete search space. If the improvement in one criterion implies the worsening in another objective, as happens with the constrained mass and the number of different cross-section types, which are our two minimising objective functions, a multiobjective optimization is required. Because of the requirement of a global, discrete and multiobjective optimization method, the evolutionary multicriteria optimization methods are suitable. Among the most recent algorithms, those which include elitism and parameter independence are outstanding. For our study, the SPEA2 [34] and NSGA-II [9] have been selected. An improved adaptation of the truncation operator in SPEA2 specially suited for two dimensional multicriteria problems, proposed in [17] is implemented. It takes advantage of the linearity of the distribution of the non-dominated solutions in the bicriteria case.

Two metrics are considered, defined on objective space, concerning about accuracy and coverage of the front. They are averaged from thirty independent runs of each algorithm.

The first metric (approximation to the front) is the M1* metric of Zitzler, representative of the approximation to the optimal Pareto front. To evaluate this metric, belonging to the scalable metrics type, the best Pareto front should be known. Its expression is [33]:

$$M1^*(U) = \frac{1}{|U|}\sum_{u \in U} \min\left\{ \| u - y \| \;\middle|\; y \in Y_p \right\} \qquad \textbf{(6)}$$

where $U = f(A) \subseteq Y$ (being A a set of decision vectors, Y the objective space and Yp referred to the Pareto set).

The second metric handles with the coverage of the front. The adopted criteria is using the number of solutions of the best non-dominated front achieved in each generation, because of the nature of our second fitness function: the number of different cross-section types, which produce a discrete non-dominated front with limited maximum number of solutions. For this reason, also there is no necessity for evaluating the smoothness of the spread of solutions along the front, because we have a discrete one, and apparently, there is no difficulty in obtaining one non-dominated solution for each number of different cross-section types between the extreme solutions of each generation.

Thirty independent runs have been considered for each algorithm (NSGAII and SPEA2) and codification case. A population size of 100 individuals and uniform crossover have been used. Three different values of the mutation rate: 0.4%, 0.8% and 1.5% are studied, comparing the binary and Gray coding.

3.1 Gray Coding for Multicriteria Optimization

The discrete nature of the search space of our problem, compound of the cross-section type of each bar, can be benefit from the discrete coding of binary coding, opposed to a less properly real coding in this particular case.

Traditionally, the more number of schemata per information unit among all the possible codifications has been claimed for binary coding [12], because of its low cardinality, being beneficial for the building blocks propagation. However, the implicit parallelism of genetic algorithms is not exclusive of the binary representation, existing for other alphabetic cardinalities of codification.

However, Binary coding suffers from not being homogeneous respect to its decimal numeric equivalent, which is normally used in its decoding. For example, the number 7 is followed by 8, but their binary representations are respectively 0111 and 1000, where every allele diverges from another. This is known as Hamming Cliff. In the phenotypic space both are consecutive values, but in the genotypic space both differ completely. It seems to be desirable a representation that maintains analogous smoothness in the phenotypic and genotypic spaces.

A Gray code is defined as a representation with 1s and 0s that permits a bijective equivalence between phenotype and genotype for consecutive integers differing only in one bit between them. It can be seen in table 1 a comparison between binary and Gray codes. It has been remarked in *italics* the values of the differing bits between two consecutive integers. It can be observed how in the case of Gray code, only one bit differ, but a more chaotic behaviour of the binary code.

Experimental studies over many kind of single objective test functions widely used in genetic algorithms show an improved behaviour of using Gray code versus the standard binary code [2][22][25].

There is an analogous theorem of the 'No Free Lunch' Theorem [32] of direct application in the comparison of binary / Gray representation [26]: 'All algorithms are equivalent compared over all possible representations'. Whitley [30] claims in this case as an example of contradiction between theory and practice, and shows that the coding influences the number of optimum that the phenotype generates for the same genotype. Gray coding can reduce this number of optima. However, it does not imply necessarily that with fewer optima the problem is solved easier, as is shown in [3]: a coding with greater optima can have less expected convergence time. There is also shown that the efficiency of the coding is dependent of the search operators. Gray coding is still an open question, as can be seen recently in [27].

Table 1. Gray Code versus Binary Code in a 4 bit string for structural optimization

Binary Code				Equivalent Integer	Gray Code			
Area cm^2	Moment of inertia cm^4	IPE	String		String	IPE	Area cm^2	Moment of inertia cm^4
7.64	80.1	80	0000	0	0000	80	7.64	80.1
10.3	171	100	000*1*	1	000*1*	100	10.3	171
16.4	541	140	00*10*	2	00*1*1	120	13.2	318
13.2	318	120	001*1*	3	001*0*	140	16.4	541
33.4	2770	220	0*100*	4	0*1*10	160	20.1	869
28.5	1940	200	010*1*	5	011*1*	180	23.9	1320
20.1	869	160	01*10*	6	01*0*1	200	28.5	1940
23.9	1320	180	011*1*	7	010*0*	220	33.4	2770
116.0	48200	500	*1000*	8	*1*100	240	39.1	3890
98.8	33740	450	100*1*	9	110*1*	270	45.9	5790
72.7	16270	360	10*10*	10	11*1*1	300	53.8	8360
84.5	23130	400	101*1*	11	111*0*	330	62.6	11770
39.1	3890	240	1*100*	12	1*0*10	360	72.7	16270
45.9	5790	270	110*1*	13	101*1*	400	84.5	23130
62.6	11770	330	11*10*	14	10*0*1	450	98.8	33740
53.8	8360	300	111*1*	15	100*0*	500	116.0	48200

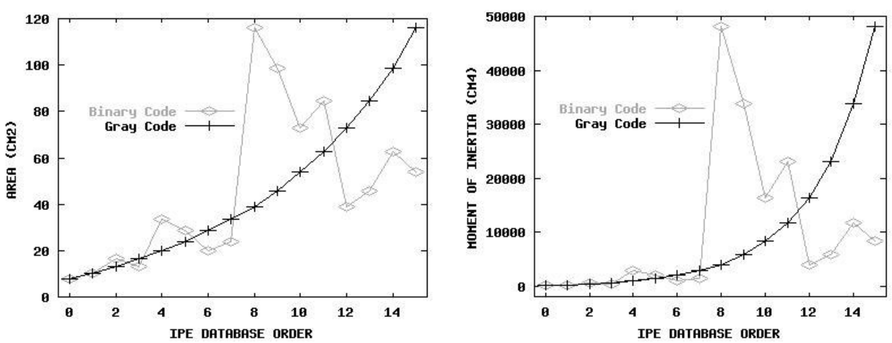

Fig. 1. Area and Moment of Inertia of IPE cross-section type database order using binary and Gray codes.

Gray coding is not unique, and there are as many possibilities as combinations are allowed to establish a continuous mapping of the genotypic and phenotypic spaces. We have implemented the Standard Binary Reflected Gray Code [25]. The suggestion of [31] has been adopted, forming the Gray code from a right displacement of the binary vector and performing a XOR with the resulting vector and the original binary one. The resulting vector is the Gray code of the binary vector. It can be implemented in C++ language easily.

The influence of the coding in the database order and its correspondence with the included cross-section types can be viewed simultaneously in table 1 and figure 1. The above cited homogeneous correspondence between phenotype and genotype is shown, in terms of two of the geometric magnitudes of the cross-section types: area and moment of inertia of the first sixteen cross-section types of the IPE series (from IPE-80 until IPE-500). Analogous figures are obtainable for the other geometric magnitudes (modulus of section, web area, etc), and for other cross-section types series (HEB, etc).

The use of Gray coding has been proven advantageous in the single criteria optimization problem of constrained mass [16]. Here is proposed an analysis of Gray code versus binary code in a discrete frame structural multicriteria optimization problem, emphasizing its use in multiobjective optimisation.

4 Test Case

4.1 Description

The test case is represented in figure 2, based on a reference problem in [20]. The figure includes the numbering of the bars, and the precise loads in Tons. Moreover, in every beam there is a uniform load of 39945 N/m. The lengths of the beams are 5.6 m and the heights of the columns are 2.80 m. The columns belong to the HEB cross-section type series, and the beams belong to the IPE cross-section type series; being the admissible stresses of 200 MPa. and 220 MPa. respectively. The maximum vertical displacement in the middle point of each beam is established in $l/300 = 1.86 \cdot 10^{-2}$ m. The density and elasticity modulus are the typical values of steel: 7850 kg/m^3 and $2.1 \cdot 10^5$ MPa., respectively.

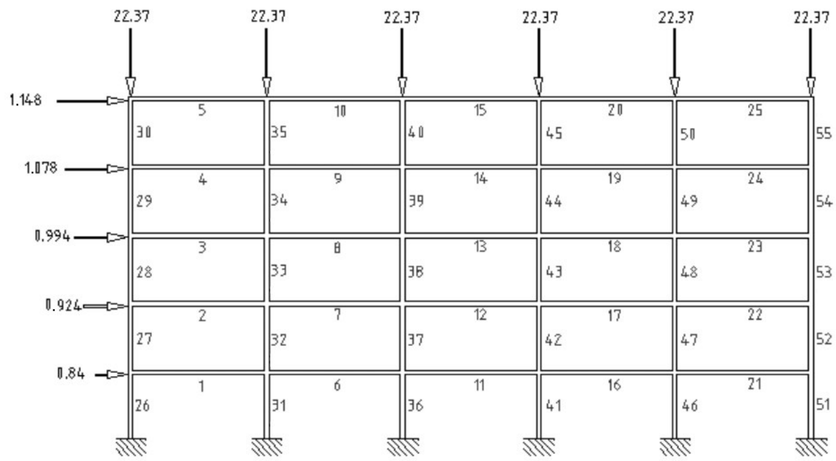

Fig. 2. Frame Test Case

4.2 Optimal Solutions

The cross-section type database is composed of the first sixteen IPE cross-section types (from IPE-80, 100, 120, 140, 160, 180, 200, 220, 240, 270, 300, 330, 360, 400, 450 to IPE-500) for the beams and the first sixteen HEB cross-section types (from HEB-100, 120, 140, 160, 180, 200, 220, 240, 260, 280, 300, 320, 340, 360, 400 to HEB-450) for the columns. Extended to the 55 bar-sized test case (and four bits per bar), this implies a search space of $2^{55*4} = 2^{220} \approx 1.7 \cdot 10^{66}$. The optimum Pareto Front is not known for this problem, and we report here the best front we have found so far, which is the reference for the metrics comparison. It is represented in figure 3, and its solutions are detailed in table 2 (detailed numerical values of solutions), table 3 (cross-section types of columns) and table 4 (cross-section types of beams).

Table 2. Pareto Set Values of Test Case

Constrained Mass (kg) (F1)	10130.04	10212.07	10318.95	10517.65	10865.28	11394.45
Constraint (kg)	2.75	5.65	0.00	0.00	213.55	272.13
Mass (kg)	10127.29	10206.41	10318.95	10517.65	10651.73	11122.32
Number of different CST (F2)	8	7	6	5	4	3

Table 3. Pareto Set Detailed Cross-Section Types (CST) of Columns of Test Case

Number of different CST (F2)	8	7	6	5	4	3
Bar nº 26	HEB160	HEB160	HEB160	HEB160	HEB220	HEB220
Bar nº 27	HEB180	HEB180	HEB200	HEB200	HEB220	HEB220
Bar nº 28	HEB160	HEB160	HEB160	HEB160	HEB220	HEB220
Bar nº 29	HEB140	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 30	HEB180	HEB180	HEB200	HEB200	HEB200	HEB220
Bar nº 31	HEB220	HEB220	HEB220	HEB220	HEB220	HEB220
Bar nº 32	HEB200	HEB200	HEB200	HEB200	HEB200	HEB220
Bar nº 33	HEB180	HEB180	HEB200	HEB200	HEB200	HEB180
Bar nº 34	HEB160	HEB160	HEB160	HEB160	HEB140	HEB180
Bar nº 35	HEB120	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 36	HEB200	HEB200	HEB200	HEB200	HEB220	HEB220
Bar nº 37	HEB200	HEB200	HEB200	HEB200	HEB200	HEB220
Bar nº 38	HEB160	HEB160	HEB160	HEB160	HEB200	HEB180
Bar nº 39	HEB140	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 40	HEB120	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 41	HEB220	HEB220	HEB220	HEB220	HEB220	HEB220
Bar nº 42	HEB200	HEB200	HEB200	HEB200	HEB200	HEB220
Bar nº 43	HEB160	HEB160	HEB160	HEB160	HEB200	HEB180
Bar nº 44	HEB140	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 45	HEB120	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 46	HEB220	HEB220	HEB220	HEB220	HEB220	HEB220
Bar nº 47	HEB200	HEB200	HEB200	HEB200	HEB200	HEB220
Bar nº 48	HEB160	HEB160	HEB160	HEB160	HEB200	HEB220
Bar nº 49	HEB140	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 50	HEB120	HEB140	HEB140	HEB160	HEB140	HEB180
Bar nº 51	HEB180	HEB180	HEB200	HEB200	HEB220	HEB220
Bar nº 52	HEB200	HEB200	HEB200	HEB200	HEB220	HEB220
Bar nº 53	HEB200	HEB200	HEB200	HEB200	HEB200	HEB180
Bar nº 54	HEB160	HEB160	HEB160	HEB160	HEB220	HEB180
Bar nº 55	HEB200	HEB200	HEB200	HEB200	HEB200	HEB220

Table 4. Pareto Set Detailed Cross-Section Types (CST) of Beams of Test Case

Number of different CST (F2)	8	7	6	5	4	3
Bar nº 1	IPE330	IPE330	IPE330	IPE330	IPE300	IPE300
Bar nº 2	IPE330	IPE330	IPE330	IPE330	IPE300	IPE300
Bar nº 3	IPE330	IPE330	IPE330	IPE330	IPE300	IPE300
Bar nº 4	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 5	IPE330	IPE330	IPE330	IPE330	IPE300	IPE300
Bar nº 6	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 7	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 8	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 9	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 10	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 11	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 12	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 13	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 14	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 15	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 16	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 17	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 18	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 19	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 20	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 21	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 22	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 23	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 24	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300
Bar nº 25	IPE300	IPE300	IPE300	IPE300	IPE300	IPE300

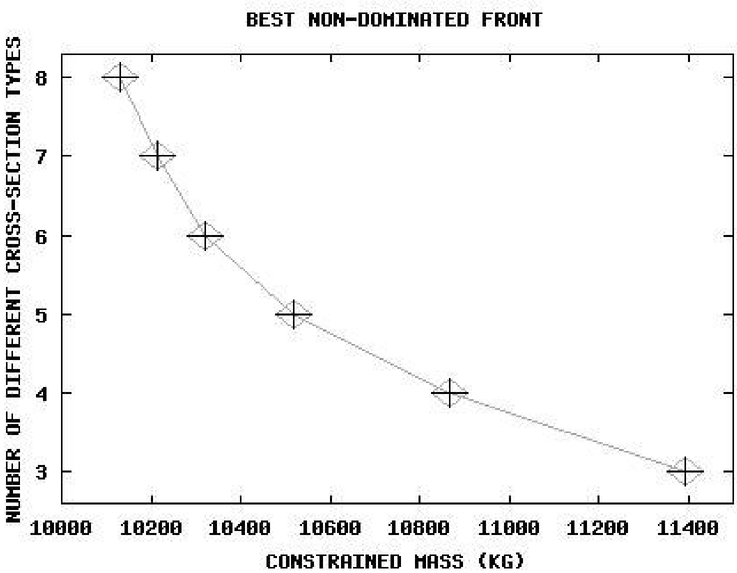

Fig. 3. Solution and Pareto Sets of Frame Structural Test Case

5 Results and Discussion

The obtained results for the SPEA2 are represented in figures from 4 to 9, where the black line corresponds to the Gray code and the grey line corresponds to the binary code. The left graph of each figure shows the mean over 30 independent runs and the right graph represents the typical deviation of the metric. Because in multiobjective optimization the search has to cope with multiple requirements, figures are organized in two groups, in order to analyse the effects of Gray code independently: approximation and coverage to the optimum front. Figures from 4 to 6 resume the results of the convergence to optimum front metric (varying mutation rate from 0.4% to 1.5%), whereas figures from 7 to 9 resume the results of the coverage front metric (varying mutation rate from 0.4% to 1.5%), as described in section 3. The x-axis corresponds to the number of evaluations of the fitness function. Also some results of the NSGAII have been included, for the mutation rate of 0.4% (figures 10 and 11).

5.1 Analysing the Approximation to the Optimum Front Metric

The nearest the value of this metric is to zero, the better. From the observation of figures 4 to 6, it can be seen a common behaviour for all the mutation rates tested. The Gray code mean metric achieves not only a better final value in all the cases (approximately half of the binary code metric value), but also during the whole convergence process it shows lower values and an initial steeper slope. It also outperforms the binary code with lower typical deviations in all cases. The variation of the mutation rate does not seem to alter the performance of this metric mean for both codings, whereas affects the performance of the typical deviation of the binary one, increasing it with the increase of the mutation rate. Similar qualitative results are also obtained using the NSGAII algorithm, as can be seen in figure 10.

5.2 Analysing the Coverage of the Front Metric

If we observe figure 3, where the optimum reference front is shown, it is noticeable that the best value of the coverage of the front using the metric described in section 3 is 6. So, the nearest the final value of the metric is to 6, the better. From the observation of figures 7 to 9, it can be seen a common behaviour for all the mutation rates tested. In the initial evaluations, high oscillations are present in this metric, which decrease significantly from 20000 evaluations. The Gray code metric achieves a final better value (around 4, versus binary, whose value is below 3.5) in all the cases, indicating a better coverage of the best non-dominated front (the most difficult solutions to reach are the right lower ones represented in figure 3). Its value is also greater during the whole convergence process, what means a wider front that increases the diversity of the non-dominated solution set. The typical deviation of Gray code is also smaller than the binary code metric, showing an enhanced robustness. The variation of the mutation rate does not seem to alter significantly the Gray code performance of this metric mean, whereas affects the binary one. It is noteworthy that similar qualitative results are also obtained using the NSGAII algorithm, as can be seen in figure 11.

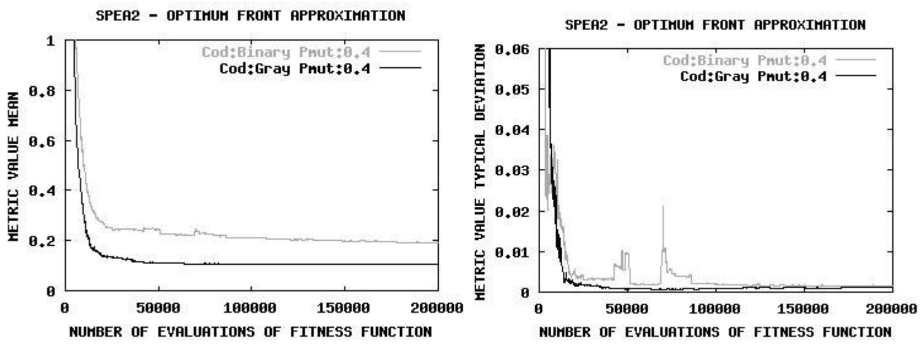

Fig. 4 Metric Approximation to Optimum Front: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 0.4%.

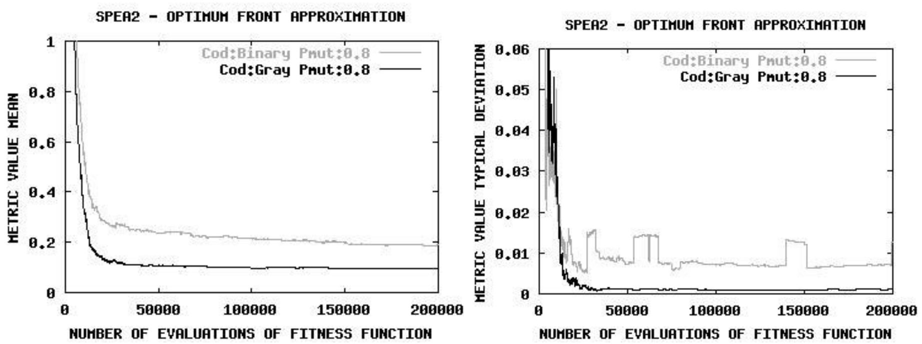

Fig. 5 Metric Approximation to Optimum Front: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 0.8%.

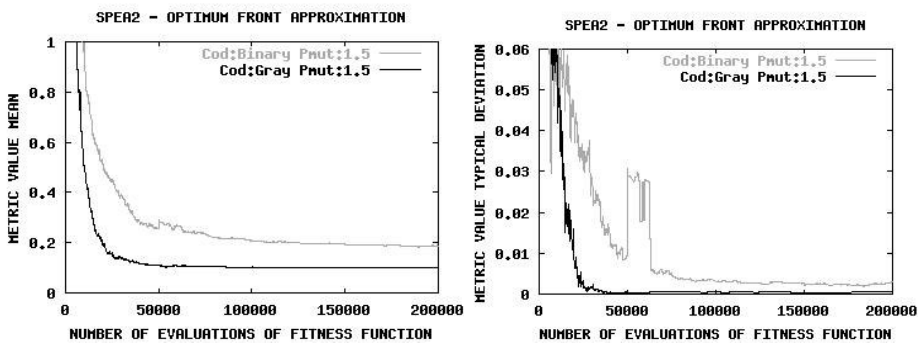

Fig. 6 Metric Approximation to Optimum Front: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 1.5%.

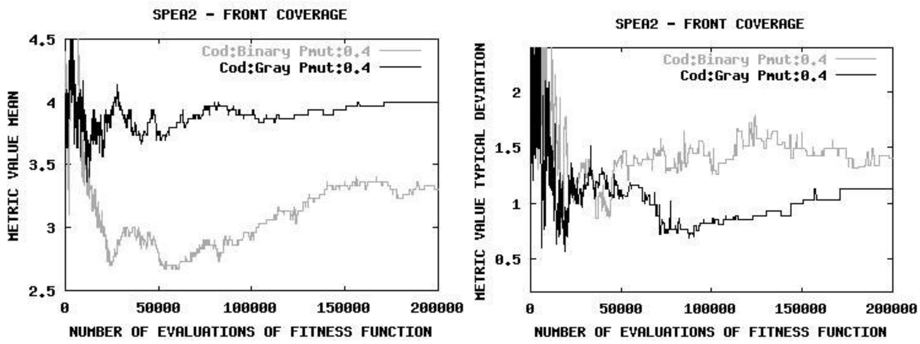

Fig. 7 Metric Front Coverage: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 0.4%.

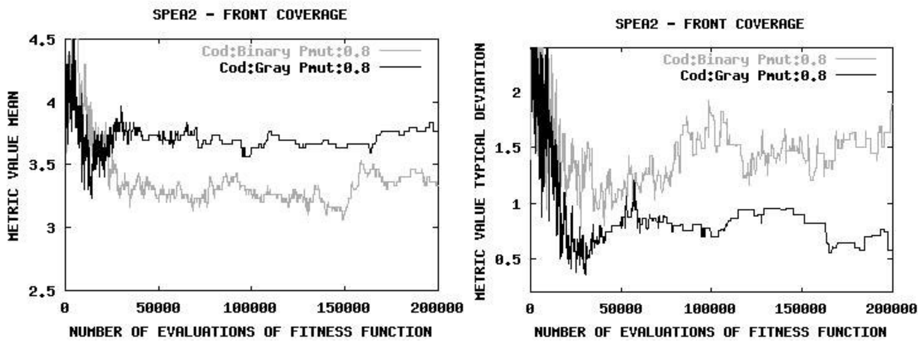

Fig. 8 Metric Front Coverage: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 0.8%.

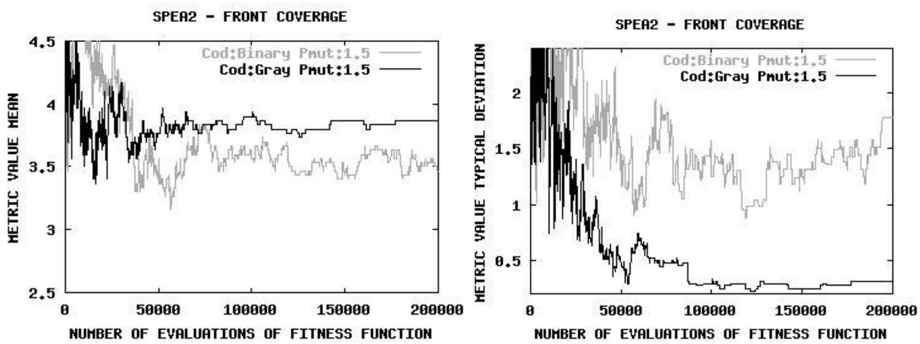

Fig. 9 Metric Front Coverage: Mean and Typical Deviation over 30 independent runs of SPEA2 and Mutation rate 1.5%.

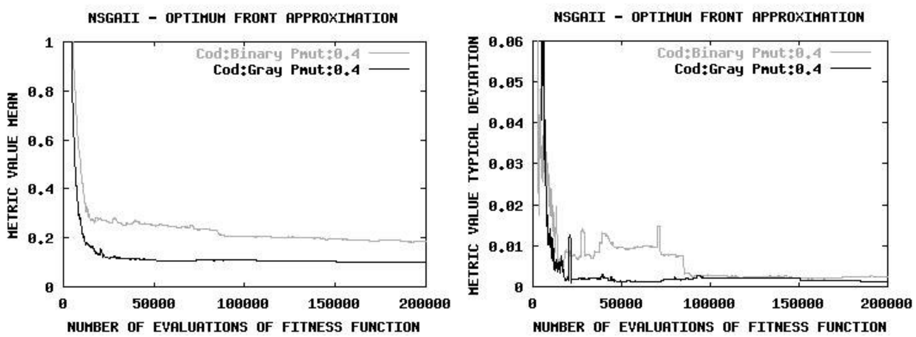

Fig. 10 Metric Approximation to Optimum Front: Mean and Typical Deviation over 30 independent runs of NSGAII and Mutation rate 0.4%.

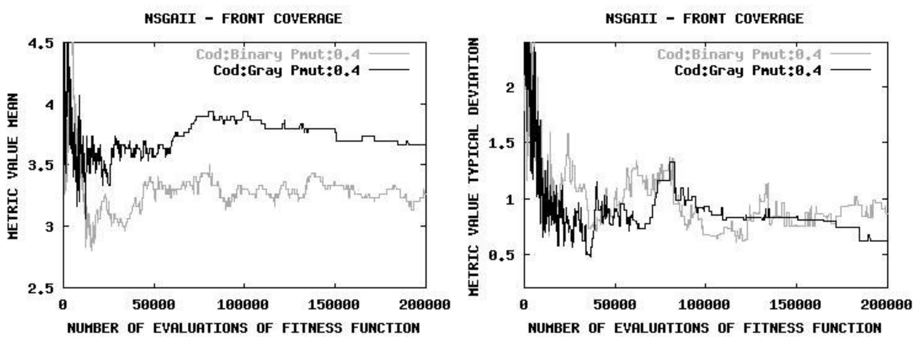

Fig. 11 Metric Front Coverage: Mean and Typical Deviation over 30 independent runs of NSGAII and Mutation rate 0.4%.

6 Conclusions

It has been analysed a discrete frame structural multiobjective optimisation problem with the point of view of the coding, through a 55-sized test case. Gray coding has been compared with binary coding considering both approximation to the optimum front and coverage of the front.

From the results exposed in section 5, it can be concluded that independently of the mutation rate used, and also of the algorithm (SPEA2 and NSGAII have been tested with similar results), the use of Gray code allows a faster approximation to the non-dominated front (more vertical slope), and more accurate (lower value of the metric), as seen in left graphs of figures 4 to 6 and 10. Moreover, the amplitude of coverage of the non-dominated front for each mutation rate used, and also for each algorithm, as seen in left graphs of figures 7 to 9 and 11, is greater when using Gray code, indicating that a more complete front is obtained. Looking at the right graphs of figures 4 to

11, where the typical deviation of the metrics are displayed, for all cases it is lower with the Gray code, revealing a higher robustness of this codification in this test case.

So, the theoretical advantages that the Gray code has due to its greater homogeneity in the correspondence between the genotypic and phenotypic spaces, and that other application studies of single criteria optimisation claimed, are also corroborated in this work by means of the obtained experimental results, in a multiobjective optimisation problem. Results show that the use of Gray code allows to obtain fast and more accurate solutions, more coverage of non-dominated fronts; both with improved robustness in frame structural multiobjective optimum design.

A generalization of this study in other multiobjective design optimisation applications, where the coding implies a phenotypic correspondence with a database ordering resulting in a more homogeneous and easier resolution, could provide more light about the empirical performance of Gray coding in multicriteria optimization.

Acknowledgements. This investigation was funded partially by the Research Project DPI2001-3570 from the Ministry of Science and Technology of Spanish Government.

References

1. S.A. Burns, Ed. *Recent Advances in Optimal Structural Design*, Institute of American Society of ASCE-SEI (2002).
2. R. Caruana, J. Schaffer, Representation and Hidden Bias: Gray vs. Binary Coding for Genetic Algorithms. In *Proceedings of the Fifth International Conference on Machine Learning*. Morgan Kaufmann (1988) pp.153-161.
3. U.K. Chakraborty, C.Z. Janikow, "An analysis of Gray versus binary encoding in genetic search", *Information Sciences: an International Journal; Special Issue: Evolutionary Computation*, 156 (2003) pp. 253-269.
4. C.A. Coello Coello, A.D. Christiansen, "Multiobjective optimization of trusses using genetic algorithms" , *Computers & Structures* 75 (2000) 647-660.
5. C.A. Coello Coello, "An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design", *PhD Thesis*, Tulane University, LA, U.S.A. (1996).
6. C. Coello, D. Van Veldhuizen, G. Lamont, *"Evolutionary Algorithms for solving multi-objective problems"*. Kluwer Academic Publishers - GENA Series (2002).
7. Cuthill, McKee, "Reducing the bandwidth of sparse symmetric matrixes*", Proceedings ACM National Conference*, New York (1969).
8. K. Deb, *Multiobjective Optimization using Evolutionary Algorithms*. John Wiley & Sons - Series in Systems and Optimization- (2001).
9. K. Deb, A. Pratap, S. Agrawal, T. Meyarivan, "A fast and elitist multiobjective genetic algorithm NSGAII", *IEEE Transactions on Evolutionary Computation* 6(2), (2002) pp. 182-197.
10. S. French, "*Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop*", Wiley, New York, 1982.
11. M. Galante, 'Genetic Algorithms as an approach to optimise real-world trusses' *International Journal for Numerical Methods in Engineering*, vol. 39, (1996) pp. 361-382.
12. D.E. Goldberg, *'Genetic algorithms for search, optimisation, and machine learning'* Reading, MA: Addison Wesley (1989).
13. D.E. Goldberg, M.P. Samtani, "Engineering Optimization via genetic algorithm". *Proceedings Ninth Conference on Electronic Computation*, ASCE, New York, NY, (1986), pp. 471-482.

14. D. Greiner, J.M. Emperador, G. Winter, "Single and Multiobjective Frame Optimization by Evolutionary Algorithms and the Auto-adaptive Rebirth Operator", in *Computer Methods in Applied Mechanics and Engineering,* Elsevier , 193 (2004) 3711-3743.
15. D. Greiner, J.M. Emperador, G. Winter, "Multiobjective Optimisation of Bar Structures by Pareto-GA", in *European Congress on Computational Methods in Applied Sciences and Engineering* (2000), CIMNE.
16. D. Greiner, G. Winter, J.M. Emperador, "Optimising Frame Structures by different strategies of GA", *Finite Elements in Analysis and Design*, Elsevier, 37 (5) pp.381-402, (2001).
17. D. Greiner, G. Winter, J.M. Emperador, B. Galván. An efficient adaptation of the truncation operator in SPEA2. In: *Proceedings of the First Spanish Congress on Evolutionary and Bioinspired Algorithms*. Eds: Herrera et al. Mérida, Spain, February 2002. (in Spanish)
18. D.E. Grierson, W.H. Pak, "Optimal sizing, geometrical and topological design using a genetic algorithm", *Structural Optimization*, 6-3 (1993) 151-159.
19. P. Hajela, C.Y. Lin, "Genetic search strategies in multicriterion optimal design", *Structural Optimization*, 4 (1992) 99-107.
20. S. Hernández Ibáñez, *'Structural Optimum Design Methods*'. Colección Seinor. Colegio de Ingenieros de Caminos, Canales y Puertos, Madrid, (1990).
21. S. Hernández Ibáñez, 'From Conventional Design to Optimum Design. Posibilities and Variants. Part I. Sensibility Analysis and local and global Optimisation', *International Journal of Numerical Methods for Calculus and Engineering Design*, (1993) CIMNE, pp. 91-110. (in Spanish)
22. R. Hinterding, H. Gielewski, T.C. Peachey, "The nature of mutation in genetic algorithms", *Proceedings of the Sixth International Conference on Genetic Algorithms* 1989, Morgan Kaufmann, pp. 70-79.
23. J. Koski, "Multicriteria optimization in structural design: state of the art", in: *Proceedings of the 19th Design Automation Conferences ASME*, (1993) pp. 621-629.
24. M. Liu, S.A. Burns, Y.K. Wen, Optimal seismic design of steel frame buildings based on life cycle cost considerations, *Earthquake Eng Struc*, 32-9 (2003) pp. 1313-1332.
25. K.E. Mathias, D. Whitley, Trasforming the Search Space with Gray Coding. *IEEE Int. Conference on Evolutionary Computation*, pp. 513-518. IEEE Service Center (1994).
26. N.J. Radcliffe, P.D. Surry, "Fundamental Limitations on Search Algorithms: Evolutionary Computing in Perspective", *Lecture Notes in Computer Science* 1000 (Computer Science Today: Recent Trends and Development), Springer, (1995) pp. 275-291.
27. J. Rowe, D. Whitley, L. Barbulescu, P. Watson, "Properties of Gray and Binary Representations", *Evolutionary Computation*, 12-1 (2004) pp. 46-76.
28. K. Sarma, H. Adeli, "Life-cycle cost optimization of steel structures", *International Journal for Numerical Methods in Engineering*, 55 (2002) 1451-1462.
29. C. Savage, "A Survey of Combinatorial Gray Codes", *SIAM Review* Vol. 39 (4) pp. 605-629 (1997).
30. D. Whitley, "A free lunch proof for Gray versus Binary Codings", *Proceedings of the Genetic and Evolutionary Computation Conference* 1999, Morgan Kaufmann pp. 726-733.
31. D. Whitley, S. Rana, R. Heckendorn, Representation Issues in Neighborhood Search and Evolutionary Algorithms, in Eds: D. Quagliarella, J. Périaux, C. Poloni, G. Winter; *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science*. John Wiley & Sons (1997) pp. 39-57.
32. D.H. Wolpert, W.G. MacReady, "No Free Lunch Theorems for Search", *Technical Report* SFI-TR-95-02-010, Santa Fe Institute, July 1995.
33. E. Zitzler, Evolutionary Algorithms for Multiobjective Optimization : Methods and Applications. *PhD Thesis*. Swiss Federal Institute of Technology (ETH), Zurich 1999.
34. E. Zitzler, M. Laumanns, L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization", *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, CIMNE, pp. 95-100 (2002).