This is an electronic reprint of the original article.
This reprint *may differ* from the original in pagination and typographic detail.

Author(s): Greiner, D.; Periaux, Jacques; Emperador, J.M.; Galván, B.; Winter, G.

Title: Nash evolutionary algorithms : Testing problem size in reconstruction problems in frame structures

Year: 2016

Version:

NASH EVOLUTIONARY ALGORITHMS: TESTING PROBLEM SIZE IN RECONSTRUCTION PROBLEMS IN FRAME STRUCTURES

D. Greiner[1], J. Periaux[2], J.M. Emperador[1], B. Galván[1] and G. Winter[1]

[1] Institute of Intelligent Systems and Numerical Applications in Engineering (SIANI),
Universidad de Las Palmas de Gran Canaria (ULPGC), 35017, Spain;
email: david.greiner@ulpgc.es

[2] Mathematical Information Technology Department (MIT) - University of Jyvaskyla, Finland &
International Center for Numerical Methods in Engineering (CIMNE) – Universidad Politécnica de Cataluña, Spain;
email: jperiaux@gmail.com

Keywords: Structural Design, Frame Optimization, Evolutionary Algorithms, Nash Equilibrium, Game Strategies, Structural Optimization.

Abstract. *The use of evolutionary algorithms has been enhanced in recent years for solving real engineering problems, where the requirements of intense computational calculations are needed, especially when computational engineering simulations are involved (use of finite element method, boundary element method, etc). The coupling of game-theory concepts in evolutionary algorithms has been a recent line of research which could enhance the efficiency of the optimum design procedure and the quality of the design solutions achieved. They have been applied in several fields of engineering and sciences, mainly, in aeronautical and structural engineering (e.g: in computational fluid dynamics and solid mechanics problems). Among them, Nash-evolutionary algorithms (Nash-EAs) have been recently applied in the single-objective reconstruction inverse design problem in structural engineering (aiming to obtain the structure whose maximum stresses match those stresses considered as references), with successful speed-up of the structural optimum search. Several test cases of different search space size bar structures are handled here, with bar sized structures up to 105 bar elements. Particularly, frames -bar structures with rigid nodes where bending moment and shear effort should also be taken into consideration- are handled here. Influence of the structural size in the comparative performance of Nash-EAs will be investigated and tested. The performance of Nash-EAs improves significantly the one of the standard panmictic evolutionary algorithms. According to the results shown here, this advantage is greater when the problem size increases.*

1 INTRODUCTION

Evolutionary algorithms have been successfully spread their use among computational engineering applications for optimum design in the last decades (see e.g.: [1,2,3,4]). Among the strategies to increase the efficiency of those algorithms, parallelization and hybridization with game theory concepts have been proven useful. Particularly, the use of Nash-Evolutionary Algorithms (Nash-EAs) has been demonstrated as acceleration tools increasing convergence speed and/or quality of solutions in computational mechanics applications [5]. In this work, a study of performance of Nash-EAs versus standard panmictic EAs is developed, considering two structural test cases of different sizes in the reconstruction inverse problem, and comparing the influence of the number of bars (size of the problem search space).

In section 2, Nash EAs are described; the reconstruction inverse structural problem handled is explained in section 3, and the frame structural test cases are detailed and justified in section 4. This paper continues with the results and discussion in section 5, and finalizes with the conclusions in section 6.

2 NASH – EVOLUTIONARY ALGORITHMS

Nash-EAs were introduced in Sefrioui and Periaux [6] for solving computational fluid dynamics problems. They are based in hybridizing the mathematical concepts of Nash equilibrium (Nash, 1950-51) [7,8] (competitive game theory where players maximize their payoffs while taking into account the strategies of their competitors) in the evolutionary search: A set of subpopulations co-evolve simultaneously each of which deals only with a partition of the search variables. These subpopulations interact to evolve towards the equilibrium; when dealing with a single objective problem, a virtual Nash game approach has been applied in inverse shape optimization computational fluid dynamics and computational solid mechanics problems as an improvement technique versus the standard panmictic evolutionary algorithms. This approach has been successfully applied in the case of inverse problems where the fitness function objective is a sum of separable terms (such as the case of many shape optimization problems) [9,10].

3 STRUCTURAL PROBLEM

The aim of the structural reconstruction problem is to achieve the structure which fits most the maximum reference stresses. The optimum structural bar design is defined as a design in which some allocation of every bar in the structure has a maximum stress value as accurately equal as the maximum reference stress for that bar. Equation (1) shows the fitness function (FF) to be minimized (reconstruction problem).

$$Fitness\ Function = \sqrt{\sum_{i=1}^{Nbars} (\sigma_{MAX-i} - \sigma_{MAX-Ri})^2} \qquad (1)$$

where σ_{MAX-i} is the maximum calculated stress and σ_{MAX-Ri} the maximum reference stress, in bar i. A null value of the fitness function in the reconstruction problem means a perfect match of stresses, and a location of the aimed structural design.

4 TEST CASES

4.1 Test Case 1: 55 bar sized frame structure

A fifty-five bar sized frame structural test case [11] is shown in figure 1, particularly the case of discrete cross-section type variables is handled here. The frame structure of reference

considered is the one corresponding to IPE330 cross section type in all beams (being the search space an interval between IPE080 and IPE500) and HEB450 cross section type in all columns (being the search space an interval between HEB100 and HEB450). Details about the corresponding maximum stress in each bar are available in [12].

A two player (two subpopulations) splitting territory approach will be used. Two Domain Decomposition (DD) player territories are tested in section 5. The distribution of bars in every case are shown as follows in figure 2 (black and cyan colors indicate membership to each territory, respectively): a Bottom-Up DD and a Left-Right DD. Territories divide the variable search space in two subsets of 27 and 28 bars.
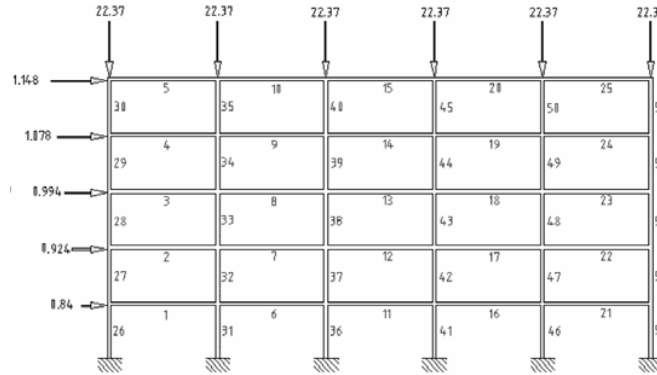

Figure 1: Computational domain, boundary conditions and loadings available in [11], 55 bar sized.
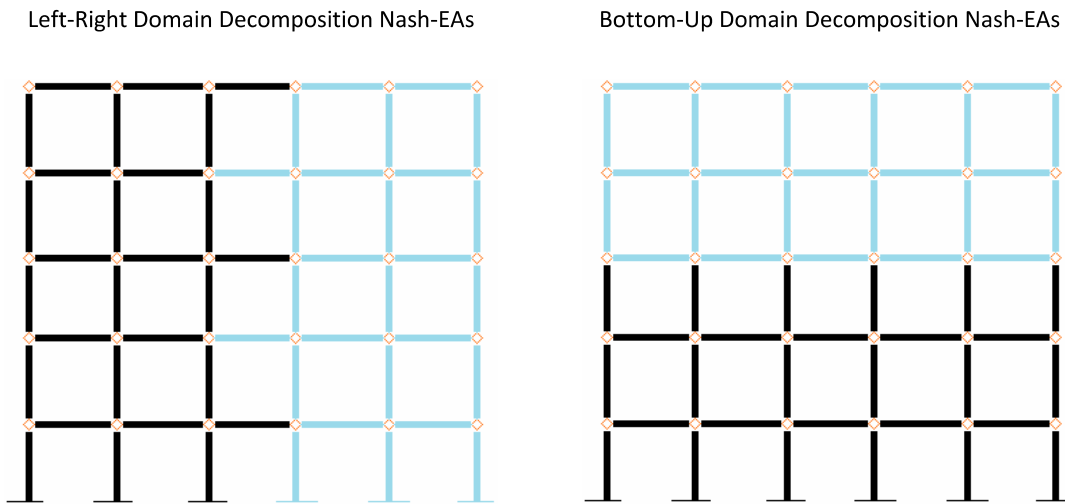
Left-Right Domain Decomposition Nash-EAs   Bottom-Up Domain Decomposition Nash-EAs

Figure 2: Nash EAs domain decompositions, 55 bar sized structure.

4.2 Test Case 2: 105 bar sized frame structure

This test case is designed purposely to study the influence of problem size in the performance of Nash-EAs versus panmictic EA. A one-hundred-five bar sized frame structural test case is shown in figure 3, particularly the case of discrete cross-section type variables is handled here. This test case is an horizontal extension of the one shown in section 4.1, with identical computational domain, boundary conditions and loadings, as well as identical cross section search space.

Also in this extended test case, a two player (two subpopulations) splitting territory approach will be used. Two Domain Decomposition (DD) player territories are tested in section 5. The distribution of bars in every case are shown as follows in figure 4 (black and cyan colors indicate membership to each territory, respectively): a Bottom-Up DD and a Left-Right DD. Territories divide the variable search space in two subsets of 52 and 53 bars.
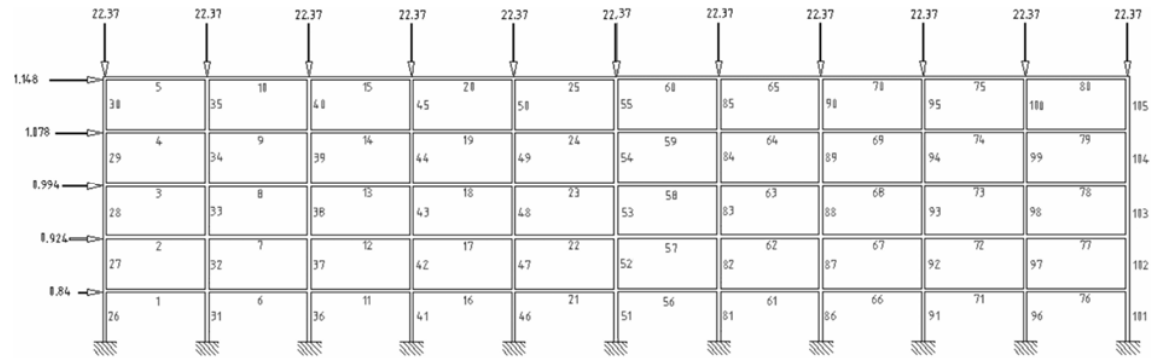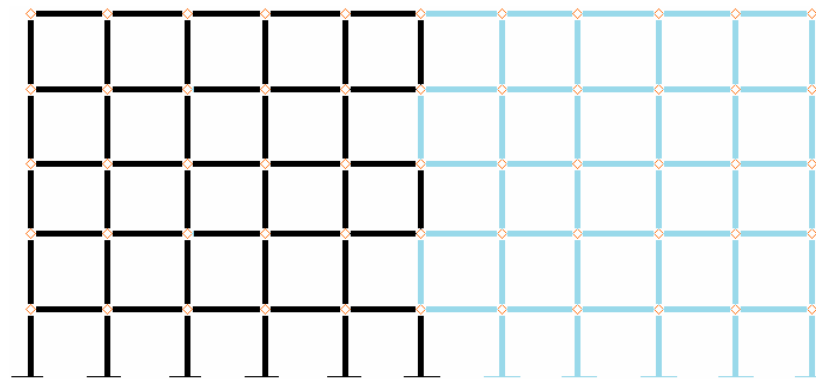

Figure 3: Computational domain, boundary conditions and loadings as in Figure 1, 105 bar sized (horizontal extension of test case 1).

Left-Right Domain Decomposition Nash-EAs
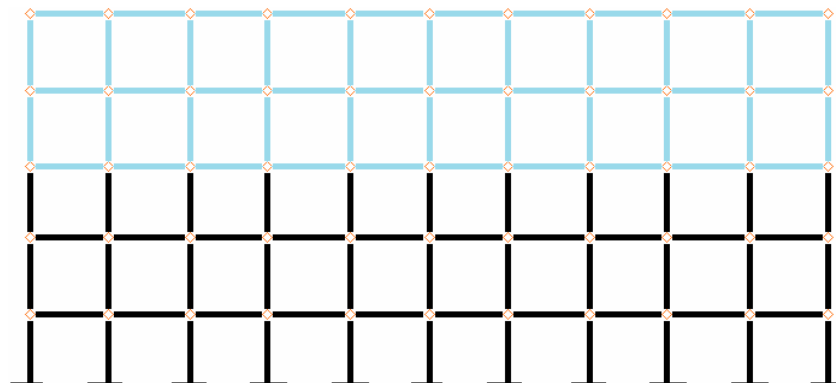

Bottom-Up Domain Decomposition Nash-EAs

Figure 4: Nash EAs domain decompositions, 105 bar sized structure.

5   RESULTS AND DISCUSSION

5.1   Test Case 1: 55 bar sized frame structure

A population size of 80 individuals, with uniform crossover, gray codification and 0.4% mutation rate are used in a set of 30 independent executions of the three evolutionary algorithms tested: standard panmictic EA, Nash-EAs with left-right partition and Nash-EAs with bottom-up partition. The stopping criterion is set as maximum number of fitness evaluations equal to $2 \cdot 10^5$.

Figure 5 shows the final average number of fitness evaluations versus the standard deviation of fitness evaluations of those executions of the three algorithms required to achieve the optimum (null) solution (all the three EAs were able to reach the optimum in all the runs in this first test case, as shown in table 1). Convergence curves of the evolution of the average, best and standard deviation values of the fitness function value are shown in figures 6, 7 and 8, respectively.
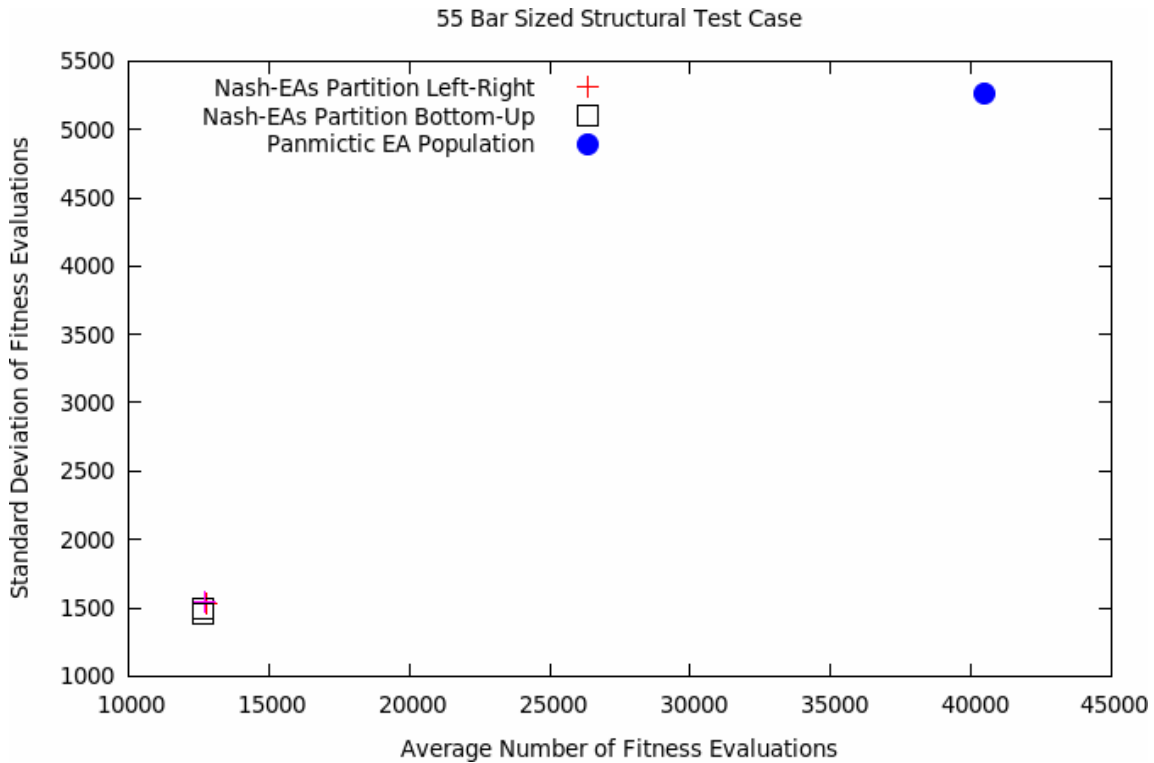

Figure 5: Panmictic EA and Nash-EAs Results - Fitness Evaluations required to obtain the optimum design out of 30 independent runs.
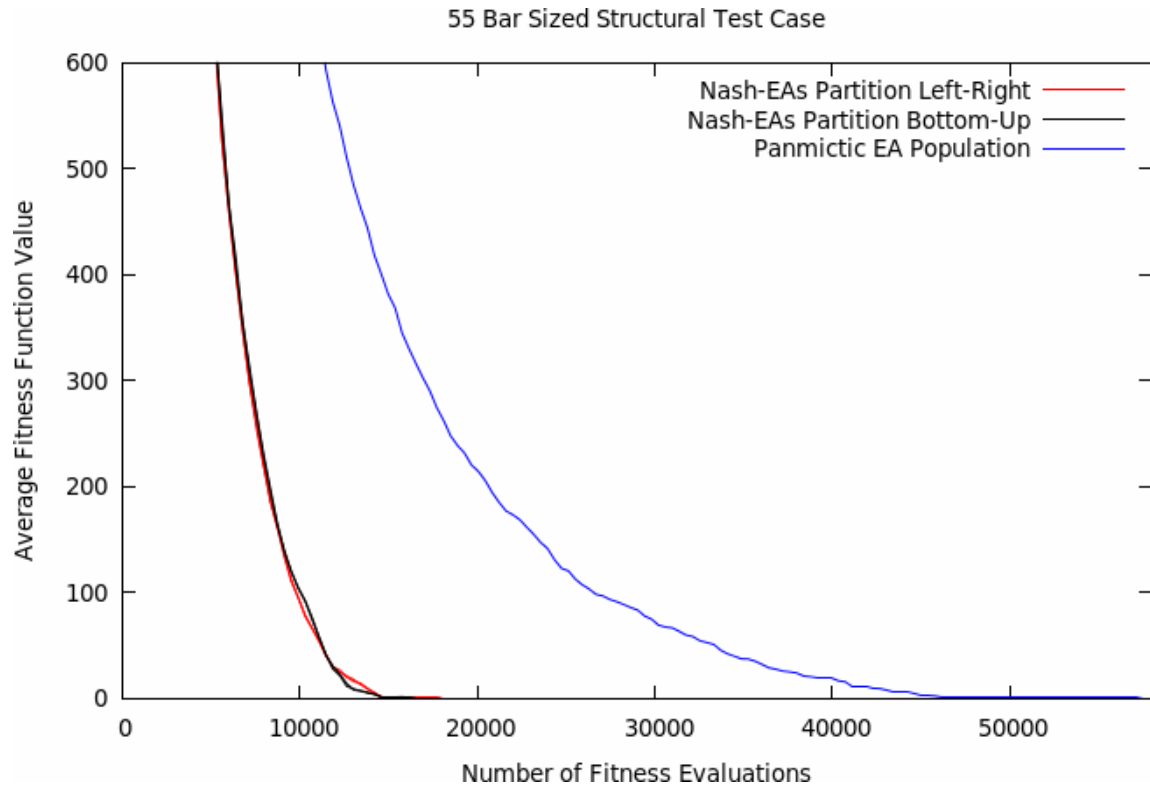
55 Bar Sized Structural Test Case

Figure 6: Average Convergence; Nash-EAs versus Panmictic EA; 55 bar sized test case
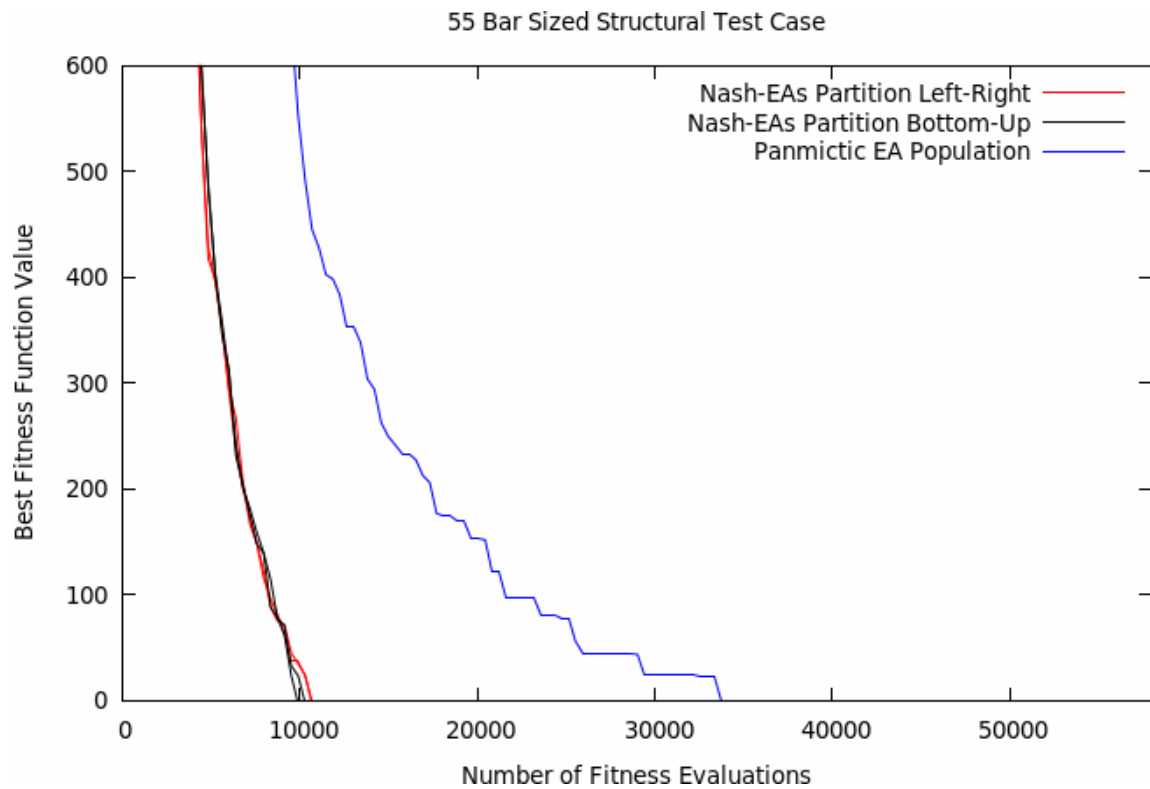
55 Bar Sized Structural Test Case

Figure 7: Best Convergence; Nash-EAs versus Panmictic EA; 55 bar sized test case
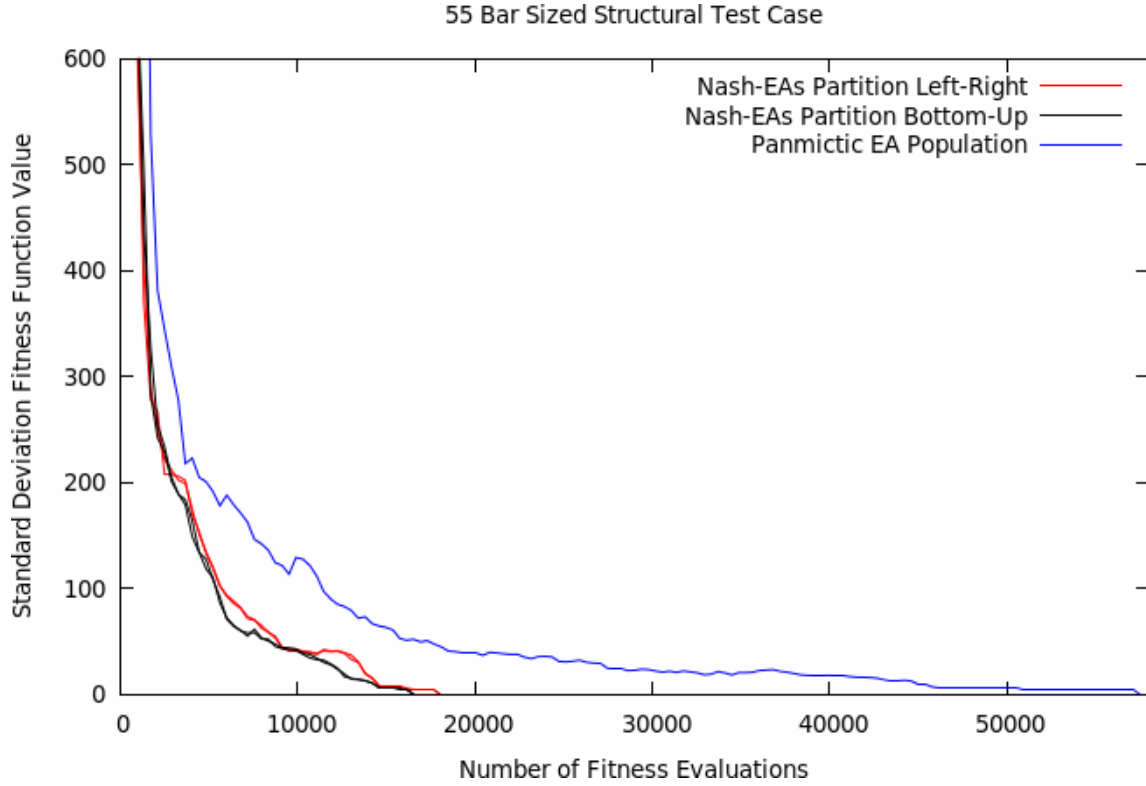
Figure 8: Standard Deviation Convergence; Nash-EAs versus Panmictic EA; 55 bar sized test case

Algorithm	Average Number Fitness Evaluations	Standard Deviation Fitness Evaluations	Best Value Fitness Evaluations	Worst Value Fitness Evaluations	Success Rate
Panmictic EA	40458	5258.8	33698	57488	**30 / 30**
Nash-EAs Left-Right Player 1	12781	1528.7	10688	18098	**30 / 30**
Nash-EAs Left-Right Player 2	12742	1536.8	10688	18098	**30 / 30**
Nash-EAs Bottom-Up Player 1	**12638**	1496.9	10298	**16538**	**30 / 30**
Nash-EAs Bottom-Up Player 2	12664	**1452.0**	9908	**16538**	**30 / 30**

Table 1: Final Results (over 30 independent runs, after a maximum number of $2 \cdot 10^5$ fitness evaluations), 55 bar sized test case.

5.2   Test Case 2: 105 bar sized frame structure

A population size of 150 individuals, with uniform crossover, gray codification and 0.4% mutation rate are used in a set of 30 independent executions of the three evolutionary algorithms tested: standard panmictic EA, Nash-EAs with left-right partition and Nash-EAs with bottom-up partition. The stopping criterion is set as maximum number of fitness evaluations equal to $2 \cdot 10^6$.

Table 2 shows the final average, standard deviation, best and worst number of fitness evaluations of those executions of the three algorithms required to achieve the optimum (null) solution, expressed as the success rate. The two Nash-EAs were able to reach the optimum in all the runs in this second test case, while the panmictic EA was completely unsuccessful (no execution was able to achieve the optimum. Convergence curves of the evolution of the average (figure 9 and zoomed figure 10), best (figure 11 and zoomed figure 12) and standard de-

viation values (figure 13 and zoomed figure 14) of the fitness function value are shown in the aforementioned figures, respectively.
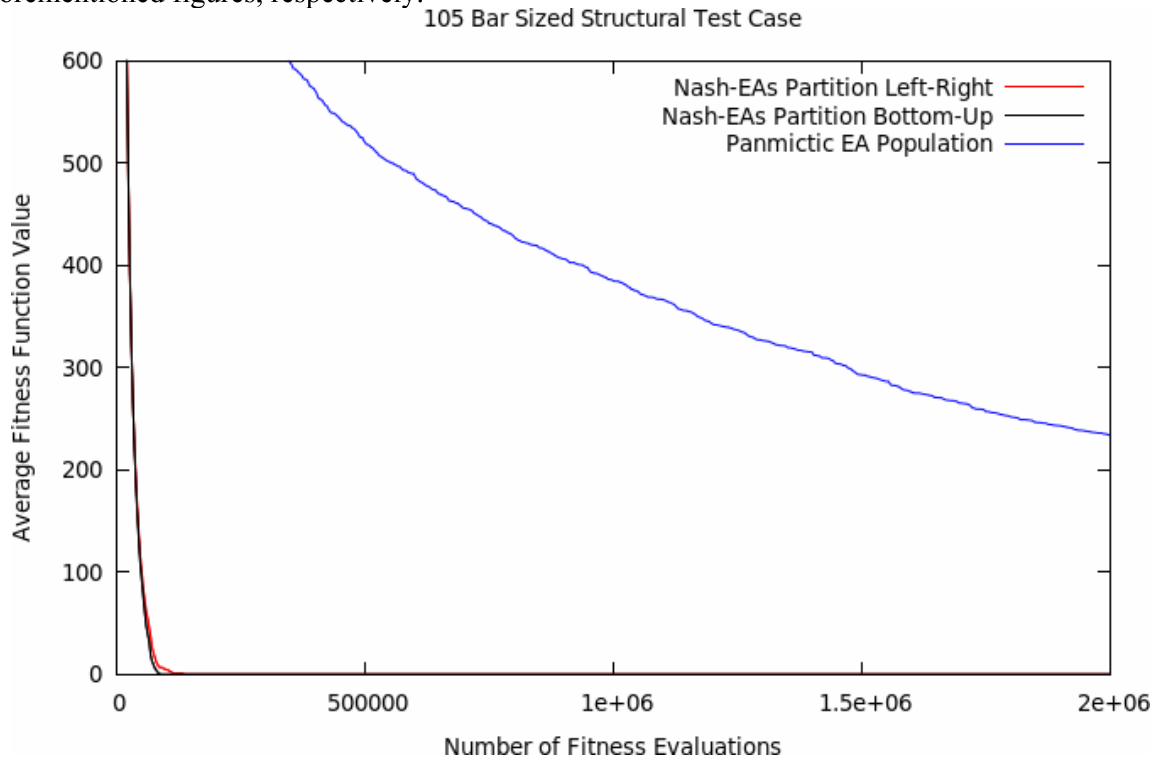

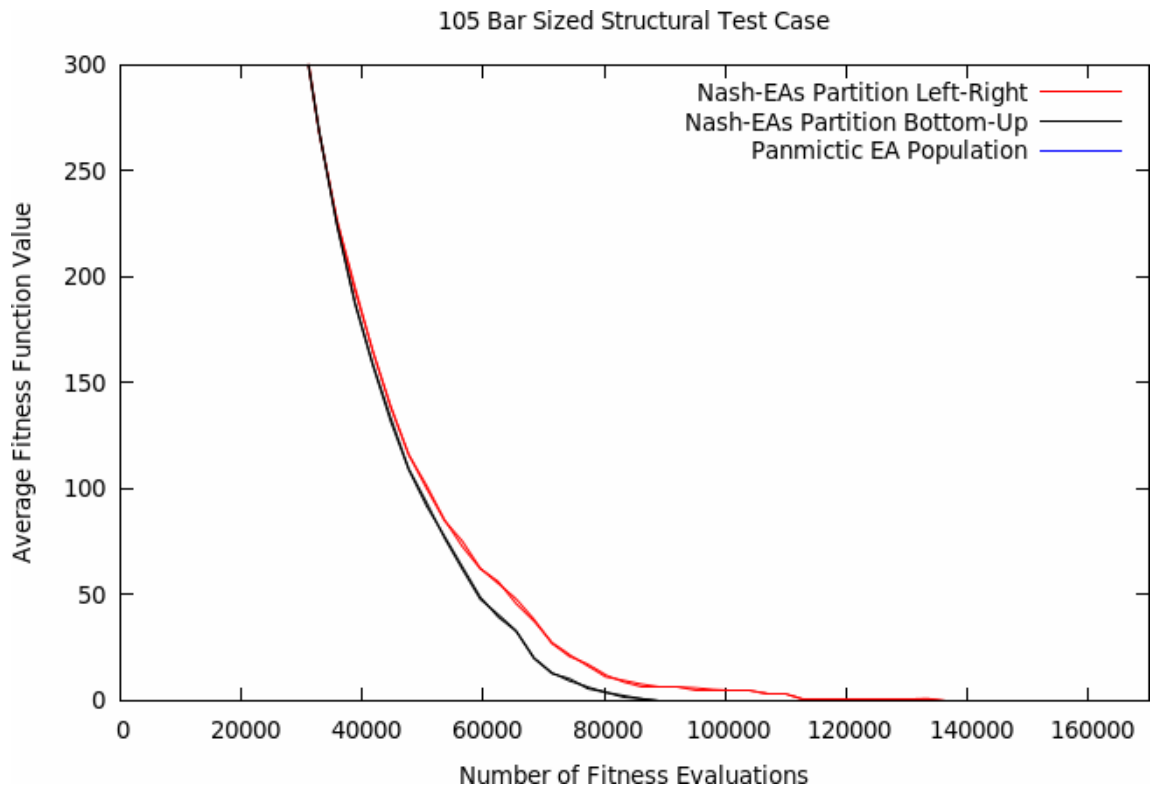Figure 9: Average Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case

Figure 10: Average Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case (Zoomed figure 9)

105 Bar Sized Structural Test Case

Figure 11: Best Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case

105 Bar Sized Structural Test Case

Figure 12: Best Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case (Zoomed figure 11)

Figure 13: Standard Deviation Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case

Figure 14: Standard Deviation Convergence; Nash-EAs versus Panmictic EA; 105 bar sized test case
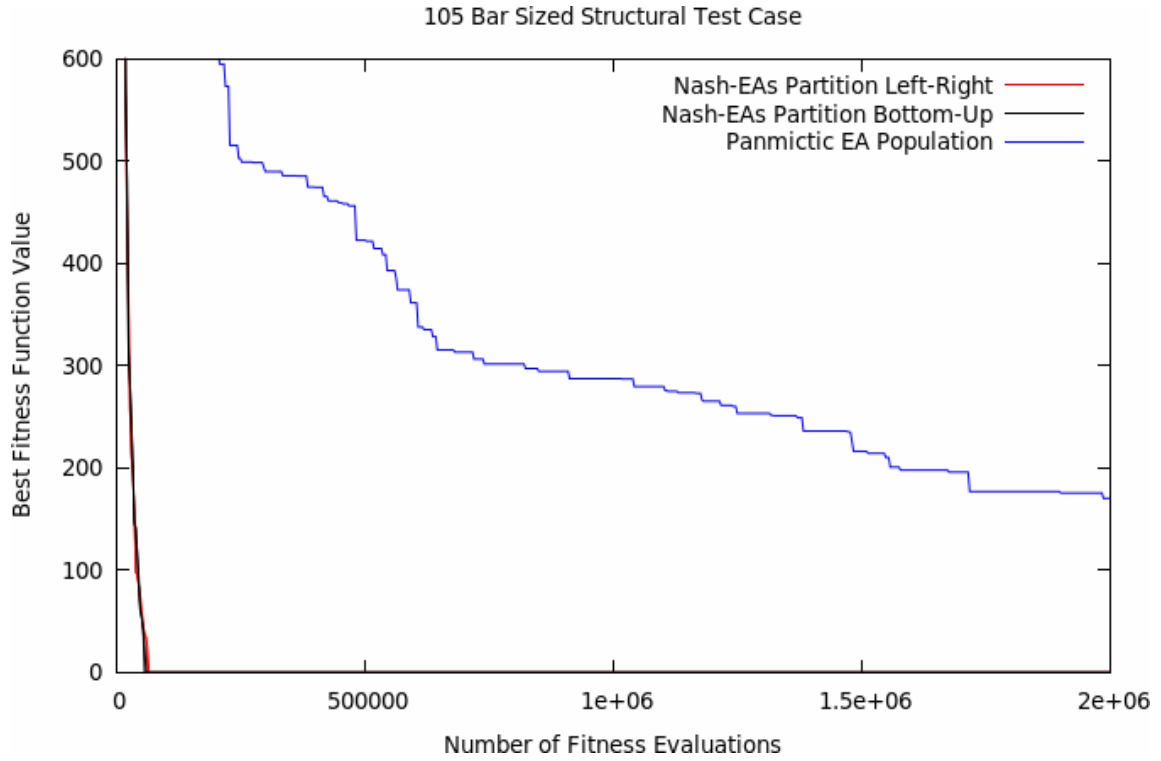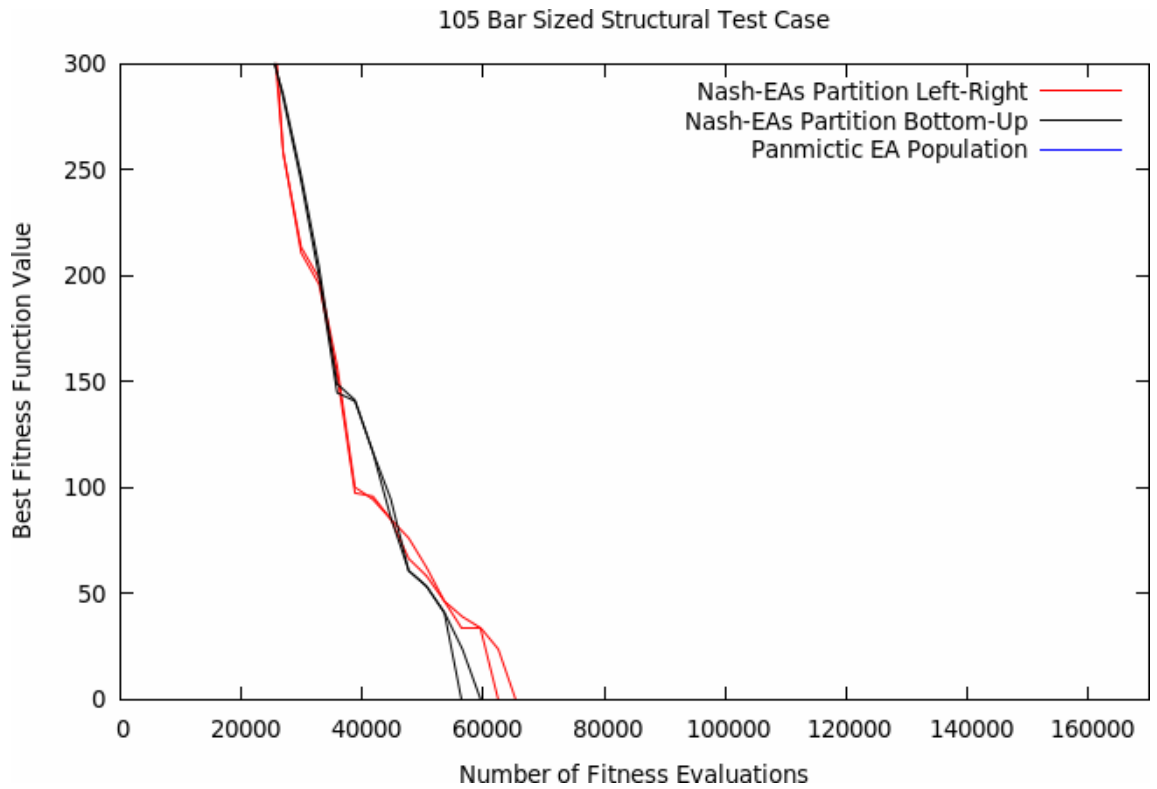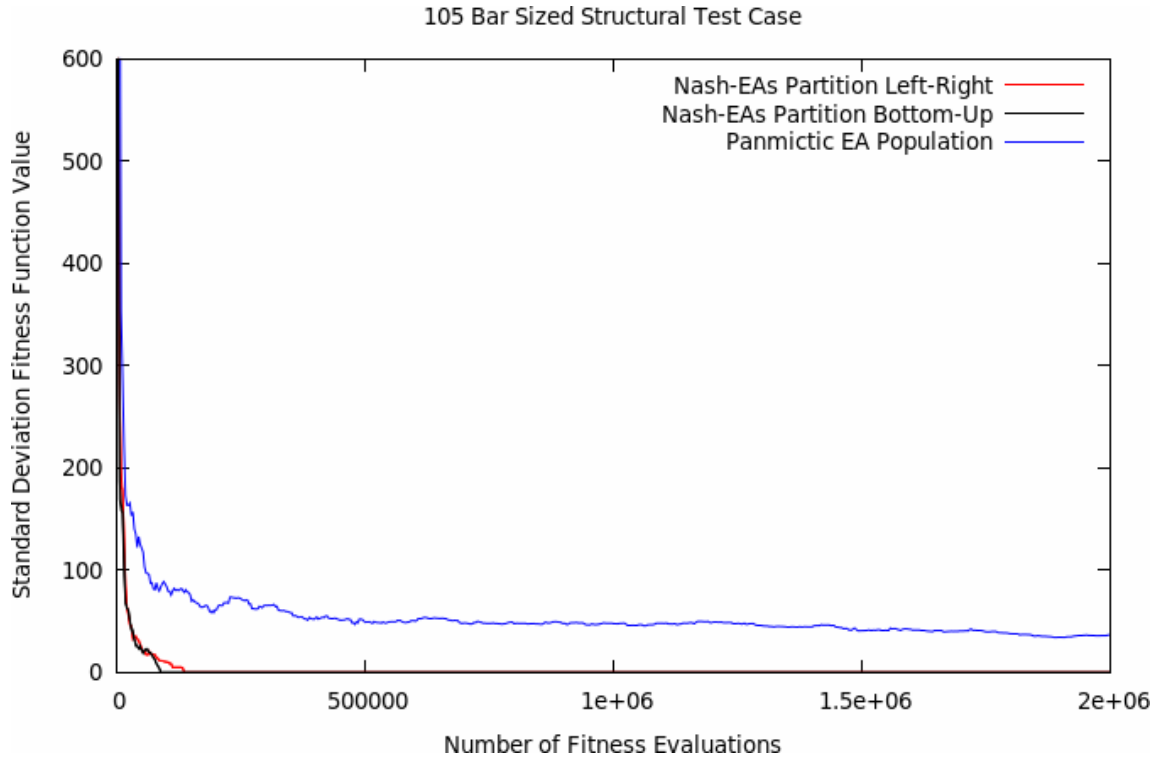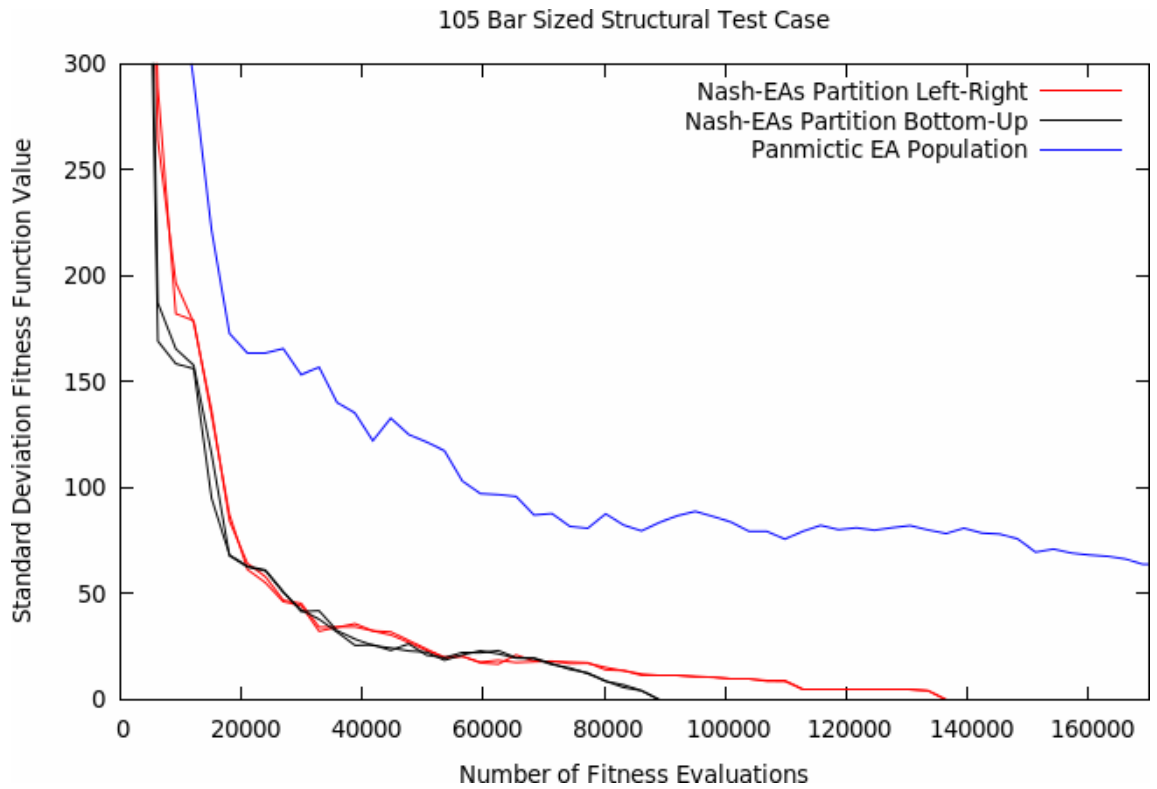(Zoomed figure 13)

Algorithm	Average Number Fitness Evaluations	Standard Deviation Fitness Evaluations	Best Value Fitness Evaluations	Worst Value Fitness Evaluations	Success Rate
Panmictic EA	Not reached	Not available	Not reached	Not reached	0 / 30
Nash-EAs Left-Right Player 1	84559.3	16617.1	65418	136458	**30 / 30**
Nash-EAs Left-Right Player 2	84658.0	16546.9	62458	136458	**30 / 30**
Nash-EAs Bottom-Up Player 1	73114.0	**7472.6**	59498	**89098**	**30 / 30**
Nash-EAs Bottom-Up Player 2	**72916.7**	7555.6	**56538**	**89098**	**30 / 30**

Table 2: Final Results (over 30 independent runs, after a maximum number of $2 \cdot 10^6$ fitness evaluations), 105 bar sized test case.

5.3   Discussion

In the smaller size test case (first one), both panmictic and Nash-EAs were equally able to reach the optimum solution consistently in all the executions. Nevertheless, Nash-EAs were much faster than panmictic EA (see table 1 and figures 6 to 8). Between the Nash-EAs Left-Right domain decomposition and the Nasb-EAs Bottom-Up domain decomposition, results are slightly better in the latter.

In the bigger size test case (second one), only Nash-EAs were able to reach the optimum solution consistently in all the executions. On the contrary, the panmictic EA was completely unsuccessful. Also Nash-EAs were much faster than panmictic EA (see table 2 and figures 9 to 14). Between the Nash-EAs Left-Right domain decomposition and the Nasb-EAs Bottom-Up domain decomposition, results are slightly better in the latter, fact that is shared with the first test case.

CONCLUSIONS

Results obtained through a reconstruction inverse problem in structural frames have evidenced that the advantage of use and efficiency of Nash-EAs versus standard panmictic EAs is greater when higher size search spaces are involved. Here particularly, two search space sizes were compared: 55 and 105 bar sized problems, which correspond to discrete search spaces of 2^{220} and 2^{420}, respectively. Robustness of the search has remained unaffected in Nash-EAs, while seriously affected in panmictic EAs.

Acknowledgements

REFERENCES

[1] N. Lagaros, M. Papadrakakis, *Engineering and Applied Sciences Optimization.* Computational Methods in Applied Sciences, **38**, Springer, 2015.

[2] D. Greiner, B. Galván, J. Periaux, N. Gauger, K. Giannakoglou, G. Winter, *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences.* Computational Methods in Applied Sciences, **36**, Springer, 2015.

[3] D. Greiner, J.M. Emperador, B. Galvan, G. Winter, J. Periaux, Optimum Structural Design using Bio-inspired Search Methods: a Survey and Applications. V. Becerra and M. Vasile eds. *Computational Intelligence in Aerospace Sciences*, Progress in Aerospace

Sciences, **244**, American Institute of Aeronautics and Astronautics AIAA, 373 - 414, 2014.

[4] D. Greiner, J.M. Emperador, G. Winter, Multiobjective Optimisation of Bar Structures by Pareto-GA. *European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS 2000*, Barcelona, España, 2000.

[5] J. Periaux, F. González, D.S.C. Lee, *Evolutionary Optimization and Game Strategies for Advanced Multi-Disciplinary Design*. Intelligent Systems, Control and Automation: Science and Engineering, **75**, Springer, 2015.

[6] M. Sefrioui, J. Periaux, Nash Genetic Algorithms: Examples and Applications. *Proceedings of the 2000 Congress on Evolutionary Computation CEC-2000*, IEEE Press, La Jolla Marriott Hotel La Jolla, California, USA, 509-516, 2000.

[7] J.F. Nash, Equilibrium points in N-person games. *Proceedings of the National Academy of Science*, **36**, 46–49, 1950.

[8] J.F. Nash, Non-cooperative games, *Annals of Mathematics*, **54(2)**, 286-295, 1951.

[9] J. Leskinen, H. Wang, J. Périaux, Increasing parallelism of evolutionary algorithms by Nash games in design inverse flow problems. *Engineering Computations*, **30(4)**, 581-600, 2013.

[10] D. Greiner, J. Periaux, J. M. Emperador, B. Galvan, G. Winter, A Study of Nash-Evolutionary Algorithms for Reconstruction Inverse Problems in Structural Engineering. D. Greiner et al. eds. *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences,* Computational Methods in Applied Sciences, **36**, Springer, 321-333, 2015.

[11] D. Greiner, G. Winter, J.M. Emperador, Single and Multi-objective Frame Optimization by Evolutionary Algorithms and the Auto-adaptive Rebirth Operator. *Computer Methods in Applied Mechanics and Engineering*, **37(35)**, 3711–3743, 2004.

[12] J. Periaux, D. Greiner, Efficient Parallel Nash Genetic Algorithm for Solving Inverse Problems in Structural Engineering. P. Neittaanmäki et al. eds. *Mathematical Modeling and Optimization of Complex Structures,* Computational Methods in Applied Sciences, **40**, Springer, 205-228, 2016.