A Study of Joint Cost Inclusion in Linear Programming Optimization

Panagiotis Armaos
Civil & Structural Engineer,
MSc Steel Construction
Iraklion, Greece
parmaos@hotmail.com

Abstract— **The concept of Structural Optimization has been a topic or research over the past century. Linear Programming Optimization has proved the most reliable method of structural optimization. Global advances in linear programming optimization have been powered to include joint cost, self-weight and buckling considerations. A joint cost inclusion scopes to reduce the number of joints existing in an optimized structural solution, transforming it to a practically viable solution. The topic of the current paper is to investigate the effects of joint cost inclusion, as this is currently implemented in the optimization code. Using IntelliFORM software, a structured series of problems were set and analyzed. The joint cost tests examined benchmark problems and their consequent changes in the member topology, as the design domain was expanding. Results are discussed. The distinct topologies of solutions created by optimization processes are also recognized. Finally an alternative strategy of penalizing joints is presented.**

Keywords-joint cost inclusion; linear programming optimization; structural optimization; IntelliFORM

I. INTRODUCTION

The term optimization is used to describe the process through which the performance of an objective function is improved toward some optimal point or points. Michell [1] was the first to suggest and develop theories concerning the optimal layout of structural frames, more than a century ago. Michell's theories were then applied in benchmark problems which where later named Michell type problems. According to [2] optimization has two distinct parts: (1) the process of improvement and (2) the optimal point. Each of them has to be studied separately. Often in optimization procedures the focus is being placed solely upon the target, the optimal point, while the interim performance is being neglected. In more complex systems with multiple objectives, reaching the optimum becomes less important. The most important goal of optimization is the search for improvement and therefore focus should be placed on methods of improvement. The objective of structural layout optimization is to minimize the volume and weight of structural members that are required to carry a specific load. With several methods being invented, linear programming optimization stands out for its accuracy and reliability.

In structural optimization the term design space is used to define the initial borders in which all possible nodes are included. The number of nodes may vary but the coordinates of the nodes or members can not extend outside this design space. Additionally all forces that are applied must be enclosed within this space. The optimum solution reached after the optimization process very often includes numerous members and consequently many nodes. That makes the optimum solution impractical for real structure design modeling. It is the scope of joint cost inclusion in optimization processes to reduce the number of actual joints associated with an optimized structural solution.

In order to give a greater picture of optimization research into finding optimum solutions, the following visualization concept is adopted. The aims of any optimization method are mainly three. Firstly, theoretical accuracy is unquestionably a goal of all optimization methods. Secondly, the size of the problems that can be solved with optimization methods needs to be large. Finally, if an optimization method is to lead to a practical design tool for structural arrangements, it should be able to incorporate realistic aspects of structural engineering design practice. In an imaginary chart each of the three aims, could be represented as an axis. The optimum point is the point where all three goals are satisfied completely. Figure 1 is the result of this visualization. The performances of linear optimization (LP) and other non-linear (Non-LP) methods are being compared. Studying the inclusion of joint costs will help increasing of the performance of linear programming optimization towards more realistically approached structural layouts.

II. THE CONCEPT OF JOINT COST INCLUSION

Solutions given from optimization analysis usually contain a very large number of nodes and members. From a constructors view, solutions like that are unmanageably complex and ineffective to be built, for several reasons. The joints of a real construction can not link more than a few members, where in the optimal solution, very often, thousands of elements may be assumed to be linked on a single node. Furthermore, optimum solutions include many short members; some of them contribute insignificantly, but can greatly increase the complexity of fabrication. The main reason though that solutions with a large number of nodes can not represent a real structure is the joint construction cost. It is widely known that the cost of constructing joints is relatively high and that increases with its complexity. Therefore, the cost of material saved in the optimization analysis can be easily outweighed by the fabrication complexity. A solution thus including a large number of joints, although optimum by weight, would be economically ineffective.

A. Introducing Joint Costs

An approach of dealing with this problem is the introduction of a penalty for every joint in the structure. Parkes [3] published a paper presenting a method, according to which joints can be taken into account in the optimization process. His suggestion was surprisingly simple and effective. In an optimization process the objective function is the volume of the structure and if the joints had to be a part of this objective function, a volume penalty had to be adjoined to them. Parkes suggested that a joint of an element is accounted as an extra length of each member, say *j* (cost of a joint) and can be represented as a joint radius *j*.

This way joints connecting many members will have large volume and therefore be excluded during the optimization process. According to the defined joint cost the solution will include more or less complicated joints. This technique implied that only a change in the member lengths should be implemented by a constant value *j*, and that complied with the linear optimization programming principle, since it included only a constant term.

This approach of penalizing joints has been adopted in IntelliFORM software, which was created by researchers of the University of Sheffield and was used for the following tests.

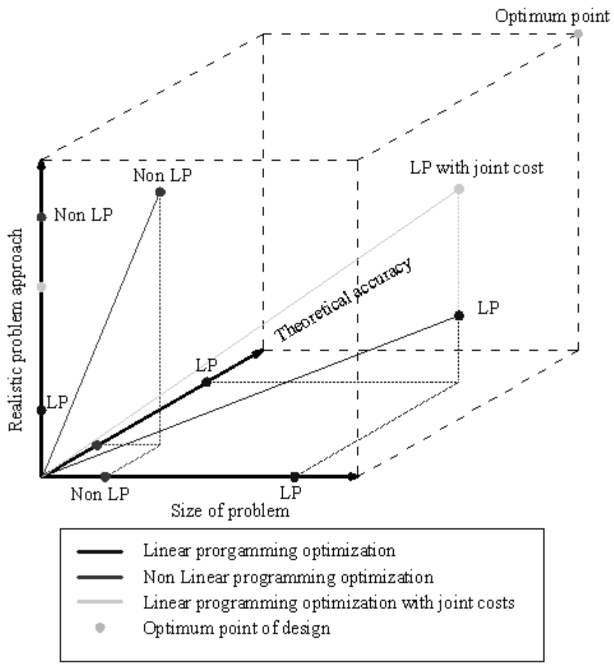


Fig. 1. Optimization goals and method performances

III. PROBLEM SERIES A: MICHELL TYPE WHEEL

A. Problem description

The first problem examined was a 2D rectangular design space with two supports (pinned and roller) at the bottom of the domain. The relative dimensions are 2 to 1 with the long dimension parallel to *x* axis. A single load was applied at the middle bottom node and had a negative unit value. The design domain is illustrated at Figure 2.

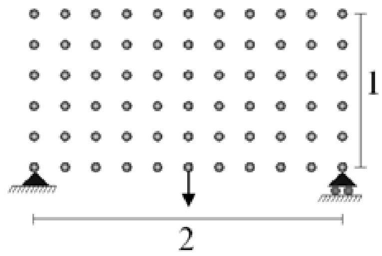


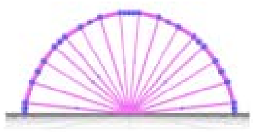
Fig. 2. Design space with 2:1 length ratio

This problem was named problem A and was studied for four design domains of increasing size. Therefore problems A_I, A_{II}, A_{III} and A_{IV} had design spaces of 20x10, 30x15, 40x20 and 60x30 nodes accordingly. The parameter joint cost *j* was examined for a typical range of values from *j*=0.0 to *j*=1,000,000. Tests were not conducted with a constant change of *j* value, since that proved to be ineffective for tracking the exact *j* value of topology change. Instead of that, tests were made much more densely in areas of possible topology changes. For example, *j* values of 0.1, 0.2, 0.4,…..0.8, 1.0 were tested and noticed that a change in topology occurred between 0.8 and 1.0. In order to track the exact *j* value that generated the change of topology, tests with more dense values in that region were conducted. Finally for value of 0.99 no topology change occurred and therefore that is assumed to occur for a *j* value of 1.0. The topology change is named 'stage'.

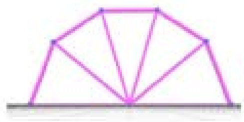
The typical topology stages of problem A_I are illustrated in Figure 3, with the joint cost value and the volume increase they correspond to. For intermediate values of joint cost *j*, the topology of the solution does not change.

The topology changes that occur as the joint cost increases are presented in Figure 4. From the chart can be seen that the relation between volume and joint cost follows a step function. On the chart, the topology layout of the solution is placed next to each step thus making clear the stages of simplification that occur. It can also be seen the slight topology change that occurs from stage (c) to (d) when the joint cost increases exponentially.

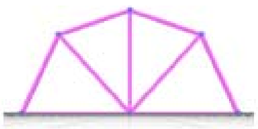
At this point the term simplicity of the structure has to be defined. The aim of joint cost inclusion is to result in optimal solutions that include fewer joints. If a solution includes fewer joints it can be considered simpler. Therefore a measure of the simplicity of a solution can be inversely the number of joints, (1/number of joints) or simply 1/*n*. Another way of representing the effect of joint cost in structural solutions is by plotting the imaginary measure of structural simplicity against joint cost. This visualization is attempted in Figure 5.



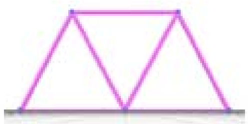
(a) *j*=0 v=3,1570852924



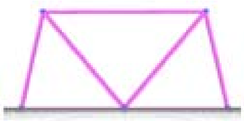
(b) *j*=1 Volume up by 3,104636%



(c) *j*=2 Volume up by 5,134993%



(d) *j*=4 Volume up by 11,0411%



(e) *j*=1000000 Volume up by 18,64438%

Fig. 3. Simplification stages of problem A_{III}

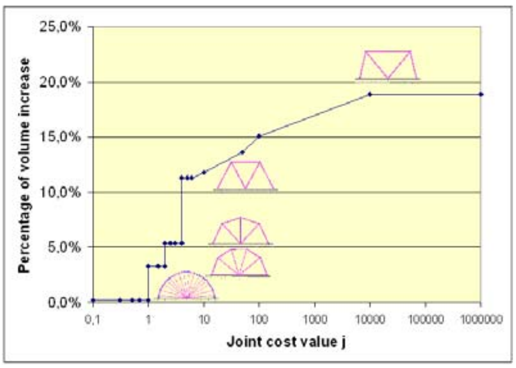


Fig. 4. Volume increase against joint cost

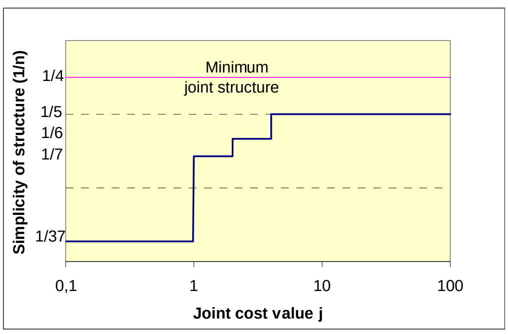
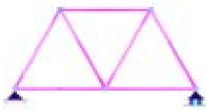


Fig. 5. Simplicity of structure against joint cost

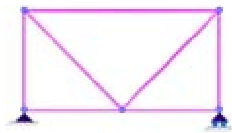
It can be seen from the previous chart that the solutions never converge to the simplest structure of four joints. Although Parkes [3] considered as the simplest structural arrangement, that of four nodes for this problem, the current joint cost criterion could not lead to this structure. This surprising finding was examined further and proved with hand calculations, and will be discussed in the next section.

B. Proof of non-convergence

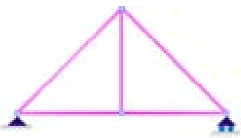
The fact that no problem solution could converge to the simplest structure of four joints required further investigation which enriched the knowledge on the actual effect of joint cost. All the models could effectively track the simplification stage or Figure 6(a), a simple truss of five joints and equal angles. As the joint cost was increasing, structural solutions illustrated a tendency towards the structural arrangement of Figure 6(b). For any infinitely large value of joint cost the arrangement of Figure 6(c) did not appear.



a)



b)



c)

Fig. 6. Structural arrangements

The equilateral truss solution (a) was calculated by hand and was found to have a volume of $V = 2\sqrt{3} = 3.4641$ for the case of all angles being 60°. Solution (b) was found having a volume of *V*= 4.0 for zero joint cost. When the joint cost formulation was included, that volume was according to (1).

$$V_{(b)} = 4 + (3 + 2\sqrt{2})\, j \qquad (1)$$

Solution (c) was also analyzed by hand calculations and surprisingly its volume was also *V*= 4.0 for zero joint cost. With joint cost that turned to equation (2).

$$V_{(c)} = 4 + (4 + 2\sqrt{2})\, j \qquad (2)$$

It can be easily noticed that the total volume of solution (b) will have a lower value than solution (c) for any joint cost *j* value. The optimization program therefore will never result in solution (c) since solution (b) will always posses lower volume. The latter contradicts with the expected simplification stages, since a structure with fewer joints is anticipated for a very high joint cost. It is inferred therefore that current joint cost inclusion is ineffective in creating always simpler topologies. An alternative joint penalty is presented further on this paper.

IV. PROBLEM SERIES B: CANTILEVER

The second problem examined with IntelliFORM was a cantilever with different lengths. With that problem series, the relation of increasing one dimension of the design space was studied. The ratio of the design space is 1/*x* with *x* taking values 0.5, 1.0, 2.0, 3.0, 4.0. A unit load was applied at the tip of the cantilever at the middle node. The vertical dimension of the design space was set to 20 nodes and the problems were as following: B_I 10x20, B_{II} 20x20, B_{III} 40x20, B_{IV} 60x20 and B_V 80x20 nodes. In Figure 7 the design space is shown.

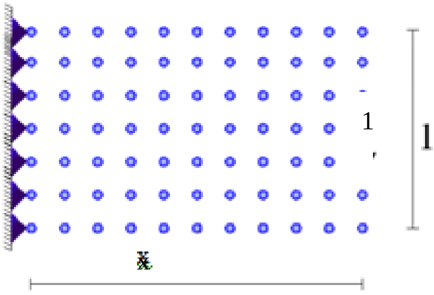
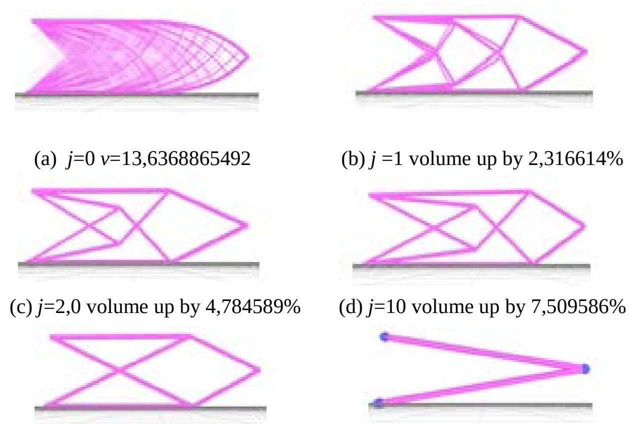


Fig. 7. Design space of problem *B*

It was noticed that type B problems always reached the simplest solution, that of three joints, for a definite value of joint cost *j*. Therefore values of joint cost problems did not have to expand to high numbers.

It is chosen that the topology simplification stages of problem B_{IV}, 20x60 nodes, are illustrated, with their corresponding volume increase, in Figure 8.



(a) *j*=0 *v*=13,6368865492 (b) *j* =1 volume up by 2,316614% (c) *j*=2,0 volume up by 4,784589% (d) *j*=10 volume up by 7,509586% (e) *j*=11 volume up by 13,6848% (f) *j*=13 volume up by 35,68689%

Fig. 8. Simplification stages of problem B_{IV}

Similarly to the previous problem the complete topology simplification sequence is illustrated at Figure 9, where next to each stage, the corresponding topology is adjoined. It was noticed that the volume increases rapidly when simpler solutions are created. The first stage eliminated a large number of nodes with a relatively small volume penalty. This is also visible in Figure 10, where the simplicity of the structure is plotted against the joint cost increase.

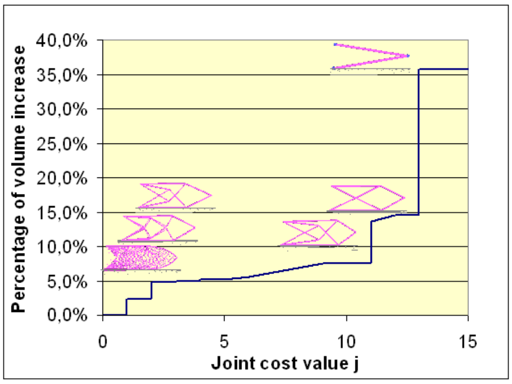


Fig. 9. Volume increase against joint cost

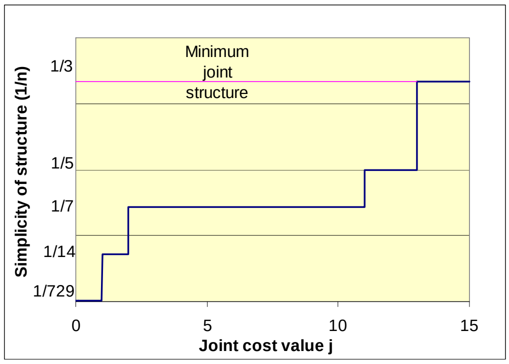


Fig. 10. Simplicity of structure against joint cost

The results of the analyses are being commented and discussed at the following chapter.

V. RESULT AND DISCUSSION

An overall assessment of the results of the problems examined came to a series of conclusions about the effect of joint cost inclusion in linear programming optimization. The current method of including joint cost into optimization is that suggested in [3] and did not always provided satisfactory results. Those along with further notes on the problems examined are listed below:

- The first topology change (stage) occurred always for a joint cost value of *j*=1.0. That was a fact for all problems, irrespectively to the geometry of the domain, support location or loads. The first stage in all the cases eliminated a large number of joints in the structure.
- No model of problems $A_{I\text{-}IV}$ could track the solution of 4 joint-triangle, Figure 6(c) as presented in [3]. This matter was further investigated in section 3.2.
- In type *A* problems the main simplification stages are three. Those stages where not always tracked by the problems. More specifically problem A_I skipped the first stage while other failed to produce a clear topology.

Instead of that, a double frame appeared as a solution, as shown in Figure 11.

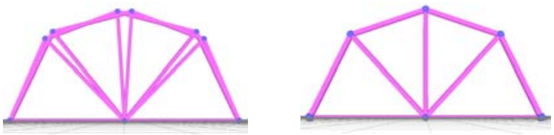


Fig. 11. Double frame appearance

That may have been caused by two reasons: either a node was not present at the desired location so a single frame could be created, or in the optimization program two distinct topologies with different number of nodes are recognized as equal. After further investigation it was found that both parameters affect the appearance of double frames. It was proven that the joint cost inclusion criterion does not actually penalize joints, but short members. Furthermore, it did not eliminate effectively the number of joints.

- Joint cost in relation to domain size: It was noticed that after the first stage which occurred for a joint cost j=1.0 later stages did not occur at the same joint cost value for all models. More specifically, for larger domains the transition of stages where shifted to higher joint costs. It was noticed that a topology stage is defined by the required volume increase of the structural arrangement rather than the value of joint cost j.
- The associated volume increase for the final simplification stage, that of three joints Figure 8(f), in the case of type B problems was significantly increased as the domain expanded horizontally. It can be inferred therefore that designing with optimization considerations can be of great benefit for large cantilever type structures as stadium roofs.

VI. Suggestion of an Alternative Joint Penalty Introduction

The joint cost as currently incorporated in linear programming optimization proved that it does not penalize effectively the number of actual joints. An actual joint penalty would allow us to track all the possible simplification stages. An alternative strategy of penalizing joints is therefore suggested in this paper.

Ideally each active joint of the structure should have an equivalent volume penalty. The difficulty of such a volume equivalence implementation in linear programming arises from the fact that in LP all nodes are supposed to be joints, fully connected. No distinction is therefore possible for the actual joints of the structure. A function is therefore required, that when implemented in the optimization process, can distinguish the actual joints and define their affect in the final optimization solution.

The function that possessed the desired behavior, modified in order to include a relative joint penalty parameter is presented in (3).

$$v_{p_{j,k}} = \frac{p_j}{e^{\left(\frac{1}{\sum_{i=1}^{n}|F_i|}\right)}} \qquad (3)$$

Where $v_{p_{j,k}}$ is the volume equivalence of joint k, F_i is the force in member i and p_j is the relative penalizing factor with positive values

The proposed volume equivalent of all joints is directly added with the volume of the structural elements and taken into account into optimization process. The total volume formulation is presented in (4).

$$V_{total} = V_{members} + \sum_{i=1}^{n} v_{p_{j,k}} \qquad (4)$$

The plot in Figure 12 shows the suggested joint penalty for different values of p_j and sums of forces. It can be seen that by altering the penalty factor p_j the influence of volume penalty can be controlled

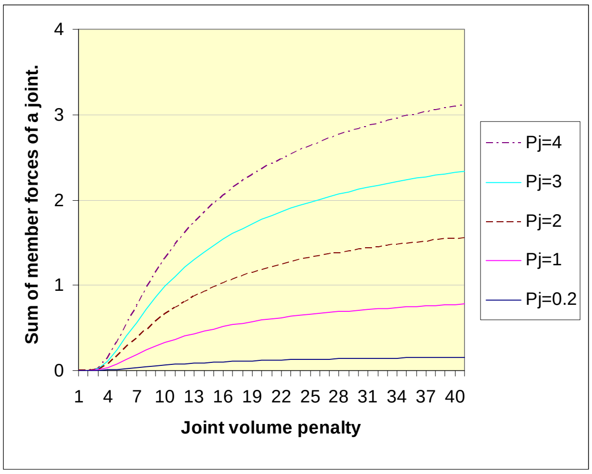


Fig. 12. Joint penalty for different values of parameter p_j

By including the suggested joint penalty into optimization processes it can be shown that the convergence discussed previously in section 3.2 can occur for a definite number of joint cost. That is depicted in Figure 13.

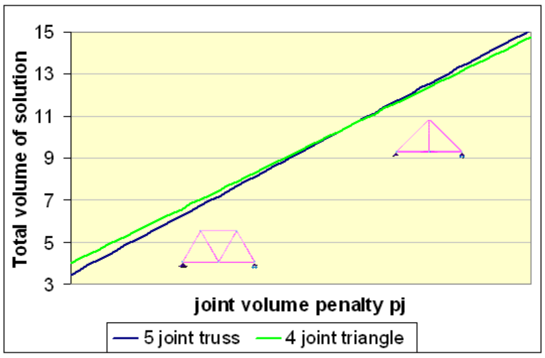


Fig. 13. Proof of convergence with suggested joint penalty

VII. Conclusions

The study of joint cost inclusion in linear programming optimization had a scope of revealing the effect of joint cost parameter *j* in different problems, as well as the simplification stages that emerge. By exploring the ways a joint cost inclusion simplifies an optimal solution, a link is created between theoretically optimum and practically buildable structures. Through this approach linear programming optimization is led towards more realistic concepts of structural solutions. However, the current way of joint cost implementing, in some occasions, proved inefficient in generating simpler solutions in the sense of decreasing joints. This in-depth study of joint cost effect in optimization process resulted in the suggestion of an alternative joint penalty integration, which illustrated inspiring results in small scale topology problems. Concluding, it is believed that with constant research and development, linear programming optimization will be able to provide a design tool to practical engineers and guide towards a more optimized structural practice.

References

[1] A. Michell, "The limits of economy of material in frame-structure", Philosophical Magazine Series 6.,Vol 8, No. 47, pp. 589–597, 1904

[2] D. E. Goldberg, Genetic algorithms in search, optimization and machine learning, Addison-Wesley, Reading, Massachusetts, 1989

[3] E. W. Parkes, Joints in optimum frameworks, International Journal of Solids and Structures, Vol. 11, No. 9, pp.1017-1022, 1975