

RESEARCH

Open Access

Numerical approach for airframe assembly simulation

Margarita V Petukhova^{1*}, Sergey V Lupuleac¹, Yulia K Shinder¹, Alexander B Smirnov¹, Sergey A Yakunin¹ and Bertrand Bretagnol²

*Correspondence:

margarita@lamm.spbstu.ru

¹Department of Applied Mathematics, Saint Petersburg State Polytechnic University, Saint Petersburg, Russian Federation
Full list of author information is available at the end of the article

Abstract

Purpose: The paper describes the methodology of contact problem solving that is used for simulation of aircraft assembly process.

Methods: The dimension of the problem is reduced after eliminating the unknowns outside the contact zone and the initial problem transforms into the problem of quadratic programming with linear constraints.

Results: The special computer code ASRP (Assembly Simulation of Riveting Process) was developed on the base of presented methodology. Now this code is utilized by Airbus. Some results of application of developed code are given in this paper.

Keywords: contact problem; riveting process; quadratic programming; dimension reduction

1 Introduction

Usually the components of the airframe (fuselage sections, wings etc.) are produced in different plants and joined together in Final Assembly Line (FAL) (Figure 1). The assembly of an airframe in FAL takes up to a week. This process is one of the bottlenecks for speed-up the overall aircraft production rate. So the optimization and acceleration of assembly process is one of the most urgent problems for aircraft building companies.

All main parts of the airframe are joined together by riveting. The assembly process is rather complicated and consists of several stages:

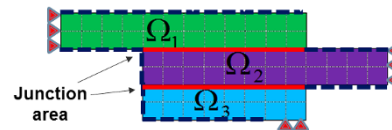
1. The parts to be assembled are brought together and initial clamping is performed.
2. The holes in assembled parts are drilled sequentially and temporary fasteners are installed in these holes. After completion of this stage the initial gap needs to be closed.
3. The temporary assembly is examined and shimming is performed if needed.
4. The temporary fasteners are removed, the assembly is taken apart, the parts are cleaned and sealant layer is applied.
5. The assembly is fastened again by temporary fasteners.
6. Sequentially the temporary fasteners are removed, the holes are widened and permanent rivets are installed in their positions.

During the assembly process it is important to control both gap between joined parts and stresses caused by installed fasteners. On the one hand tight contact between parts should be achieved; on the other hand engineers should avoid cracks, composite layer separation and part destruction.

Figure 1 Assembly of Airbus A380.



Figure 2 Considered model. Three panels (green, purple and blue ones) are joined together. Maximum possible contact zone is shown in red.



The main goal of the presented work is to develop special tool which allows engineers to perform simulations in order to evaluate displacements and stresses of aircraft parts on the assembly line. For this purpose ASRP software complex was developed for solving special class of contact problems that particularly arise during the simulation of riveting process. Let us denote this class as class R . The considered class R has specific features that were taken into account during construction of problem solving technique:

- The area where contact may occur is known apriori and is relatively small regarding whole model. Further we will refer to it as junction area.
- Tangential displacements are negligibly small in comparison with the normal ones in the junction area.
- External loads are applied in the junction area or can be transferred there.
- Friction is not taken into account.
- Stress state of each considered body obeys linear theory of elasticity.
- The problem is considered as stationary.
- Usually it is necessary to perform the number of numerical experiments (e.g. with different fastener arrangements or initial gaps) on the same mechanical model.

The developed ASRP complex considerably surpasses commercial FEM codes in speed and user convenience for simulation of aircraft assembly process.

The structure and features of this software are described in [1] and [2].

The present paper gives the detailed description of developed mathematical algorithm as well as application examples.

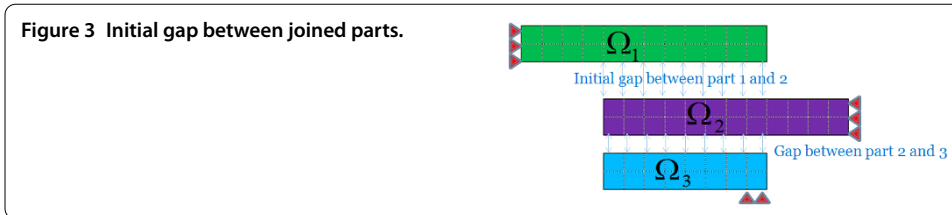
2 Main section

2.1 Methodology for gap computations

Let us consider finite element model representing several panels to be assembled (as it is shown in Figure 2). These panels overlap in so called junction area shown by red lines. The contact between panels is possible only inside the junction area.

Initially panels are separated by distance called initial gap (see Figure 3).

Let M be a number of finite element nodes in the model and $\{x_j\}_{j=1}^M$ is a set of finite element nodes. The unknowns are displacements in all the nodes $\{U_j\}_{j=1}^M$. Here we assume that the set $\{x_j\}_{j=1}^M$ includes the nodes of upper, middle and lower panels.



If we do not take into account possibility of contact between panels, the determination of $\{U_j\}_{j=1}^M$ needs only solving the system of linear equations:

$$K \cdot U = F, \tag{1}$$

where $K = \{k_{ij}\}_{i,j=1}^M$ is global stiffness matrix, $F = \{f_j\}_{j=1}^M$ is vector of loads.

Also it is necessary to impose the boundary conditions to avoid rigid body motion:

$$U_j = U_{0j}, \quad x_j \in \Gamma_{Uh}, \tag{2}$$

where Γ_{Uh} is a part of the boundary where displacements are specified.

The system (1), (2) is supplemented with following restriction to avoid interpenetration of the panels:

$$(A \cdot U)_j \leq G(x_j), \quad x_j \in \Gamma_{Ch}. \tag{3}$$

Here A is linear operator determining the relative displacement of parts in normal direction, $G(x_j)$ is a function of normal initial gap in the node x_j ; Γ_{Ch} is junction area that is the set of nodes that can come into contact.

To finalize problem statement we pose conditions on contact stresses: in the contact points the normal stresses are contradirectional and have the same value. For class R the friction between parts is not taken into account, so tangential stresses in contact points are zero. Thus we consider so-called unilateral contact problem. As it was shown in [3], the solution of this problem provides good agreement with aircraft assembly experiment.

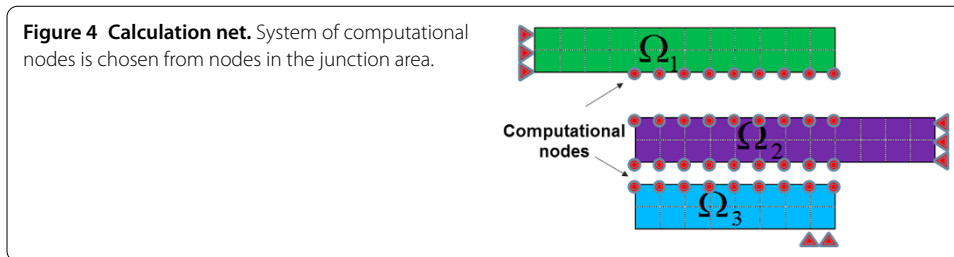
The equivalent variational formulation is used to determine the displacements. The displacement vector $\{U_j\}_{j=1}^M$ under the load $F = \{f_j\}_{j=1}^M$ provides minimum to the energy functional

$$E(U) = \frac{1}{2} U^T \cdot K \cdot U - F^T \cdot U \tag{4}$$

on the set of all admissible displacements A_C :

$$A_C = \{U \mid U = U_0 \text{ in } \Gamma_{Uh}, A \cdot U \leq G \text{ in } \Gamma_{Ch}\}. \tag{5}$$

If we solve the problem belonging to denoted class R we can reduce its dimension by eliminating all unknowns outside the junction area. It will lead to significant decrease of computational time without loss of accuracy. Similar approach of dimension reduction was used in [4]. Also we recall the condition of tangential displacement infinitesimality that allows us considering node-to-node contact model. Thus we need to compute only normal displacements of nodes in junction area.



Further in the paragraph we are to formulate the reduced minimization problem (with unknowns in junction area) equivalent to (4) and (5).

For that we choose the system of computational nodes $CN = (cn_1, cn_2, \dots, cn_l)$ in junction area, for example, as in Figure 4 (l is the total number of nodes in junction area). We denote it calculation net. It consists of nodes where displacements will be computed. We will build the reduced rigidity matrix K_C , which allows to pass from whole model to calculation net.

Reduced rigidity matrix characterizes the response of computational nodes to the applied load. To be more precise, $K_C^{-1} = \{k_{ij}\}_{i,j=1}^l$ is inverse matrix to K_C and k_{ij} is a displacement of j th node caused by the unit load applied to i th computational node. Let us consider a method for computing matrix K_C .

We divide whole displacement vector into two parts: $U = \begin{pmatrix} U_C \\ U_R \end{pmatrix}$, where U_C is vector of displacements in junction area, U_R is vector of displacements in all other nodes.

Then global stiffness matrix K from (1) can be written in blocks, and system (1) transforms to

$$\begin{pmatrix} K_{CC} & K_{CR} \\ K_{CR}^T & K_{RR} \end{pmatrix} \cdot \begin{pmatrix} U_C \\ U_R \end{pmatrix} = \begin{pmatrix} F_C \\ F_R \end{pmatrix}. \tag{6}$$

Calculating Schur complement for the block K_{RR} (see [5]) we obtain reduced rigidity matrix K_C instead of block K_{CC} .

Also this matrix can be computed using the formula:

$$K_C = K_{CC} - K_{CR} \cdot K_{RR}^{-1} \cdot K_{CR}^T. \tag{7}$$

Reduced rigidity matrix is computed once for considered junction during data preparation stage. Then this matrix is stored and can be used for series of computations because changes of initial gap or applied loads do not affect K_C . This computational decomposition is relevant for simulation of riveting process as it is aimed at solving variety of problems with different fastener configurations and different relative positions of joined parts.

Most of the research work devoted to simulation of riveting process is done for the case when the contact problem is solved only in the neighborhood of one or several fasteners, i.e. local modeling is performed. Presented methodology allows us to perform global modeling, which means to calculate displacements and contact forces taking into account that contact may occur anywhere in the junction area. Dimension reduction technique is well known as substructuring or superelement creation (for example, see [4]) and turned out to be very efficient when applied to considered type of contact problems.

Now we can formulate equivalent minimization problem for U_C :

$$\min_{U_C \in A_C} \left(\frac{1}{2} U_C^T \cdot K_C \cdot U_C - F_C^T \cdot U_C \right), \quad (8)$$

where $A_C = \{U_C \mid A \cdot U_C \leq G\}$.

The rest of displacements U_R are recovered by the following relation:

$$U_R = -K_{RR}^{-1} \cdot K_{CR}^T \cdot U_C. \quad (9)$$

It is not difficult to prove that vector $U = (U_C \ U_R)^T$ is a solution of minimization problem (4) and (5) if and only if U_C is a solution of the problem (8) and U_R is determined by equation (9).

Problem (8) is a quadratic programming problem (QPP). The method for its solution is discussed in the next section.

The solving of problem (8) gives us the field of displacements in junction area U_C as well as contact forces obtained as Lagrange multipliers. If we are interested in complete field of displacements U , we can add the relations

$$U_j = U_{Cj}, \quad x_j \in \Gamma_{Ch}, \quad (10)$$

to the system (1), (2) and solve the obtained system of linear equations. Or alternatively we add contact forces to the vector of loads F and also solve the system (1), (2) to obtain U . Knowing field of displacements, it is easy to obtain all information about deformed-stress state of our system (including strains and stresses).

Also it is possible to use more refined finite element model to find field of stresses within considered algorithm. To do so we interpolate the computed displacements and/or contact forces to the refined mesh and then compute stresses using linear static analysis.

2.2 Quadratic programming problem solving

2.2.1 Problem statement

In this section we consider the algorithm used for solving QPP (8) and its adaptation to the peculiarities of contact problems from class R .

In (8) matrix K_C is ill-conditioned. Moreover, it is symmetric and positive-definite. If we consider riveting process of n parts, then vector U_C is constructed from displacements of nodes on each part independent of each other:

$$U_C^T = (U_{C1}^T, U_{C2}^T, \dots, U_{Cn}^T), \quad (11)$$

where U_{Ci}^T is a vector of displacements in contact nodes of i th part.

For such a vector of displacements full reduced rigidity matrix takes a following appearance:

$$K_C = \begin{pmatrix} K_{C1} & 0 & \dots & 0 \\ 0 & K_{C2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & K_{Cn} \end{pmatrix}, \quad (12)$$

where K_{Ci} is reduced rigidity matrix of i th part.

From (12) one can see that matrix K_C has block diagonal structure.

Matrix A that defines admissible set in problem (8) has a simple structure: each row of matrix A contains one or two non-zero elements. It is due to the fact that node-to-node contact is considered. Thus constraint matrix A can be considered as sparse matrix.

2.2.2 Algorithm description

We made a review of algorithms for solving QPP: interior point methods, gradient projection methods and active set methods. The last group meets our goals as iteration process finishes after finite number of steps and algorithms allow to simply vary initial approximation. One of the most popular algorithms for solving QPP is described in [6, 7]. It is known as Goldfarb-Idnani algorithm with Powell modification and is used in this project (further it is called G-I-P). Goldfarb-Idnani active set method is proved to be fast and efficient; furthermore the Powell modification is favorable for ill-conditioned problems. This method is also implemented in well-known libraries such as IMSL Numerical Library and Scilab for solving quadratic programming problem.

The main idea of considered algorithm is:

- To start at the point of unconstrained minimum of the problem. If it satisfies the constraints (there is no contact in junction area), then this is desired solution.
- If the constraints are violated, the iteration procedure starts. At each iteration step the current point is forced to satisfy one of violated constraints in a way:

$$(A \cdot U_C)_i = G_i, \quad i = \overline{1, n}. \quad (13)$$

- It is done until the point is in admissible set and this means that solution is obtained.
- If it is not possible to satisfy any constraint during iteration step, rollback is performed: point is shifted in such a way that one of the current active constraints is violated again.

It is worth mentioning that iteration step is made in both primary and dual spaces (not only the point is shifted but also Lagrange multipliers).

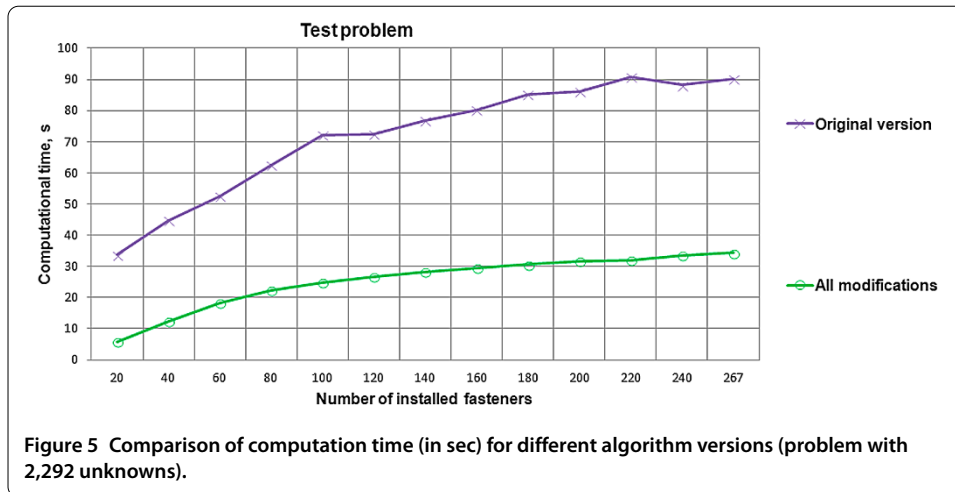
According to described procedure it can be clearly seen that solution can be found faster if less nodes in contact zone are involved into contact interaction. The greater contact area is, the more steps are required to satisfy each constraint, the longer time algorithm works.

2.2.3 Implemented modifications

In order to speed up the computations we propose several modifications of described algorithm that make use of input data structure given in Section 2.2.1 and take into account algorithm specifics.

Implemented modifications are:

1. Making use of the block structure of matrix K_C .
2. Making use of simple structure of matrix A . The number of non-zero elements in matrix A is not greater than $2n$, therefore the number of multiplication may be decreased by a factor of ten: perform $O(n)$ operations instead of $O(n^2)$.
3. Choosing the most appropriate constraint. At each iteration step the next constraint to be satisfied is being chosen. In original version of algorithm the first most violated constraint is taken into consideration. It makes sense to introduce more complicated procedure of constraint choosing. We rely on the fact that if force is applied to



computational node, then it means that fastening element is installed in this point in terms the riveting process. Usually if the fastening element is installed, the gap in this point is closed and condition (13) is fulfilled. Hereby, we are to choose the constraint that links the nodes where maximal force is applied.

4. Saving results from previous computations. When simulating the riveting process it is usually necessary to test different fasteners' configurations, i.e. to solve quite similar problems where only vector F may vary. If the fastener configuration changes slightly it is reasonable to start not from the point of unconstrained minimum but from the solution obtained for previous problem.
5. Solving the dual QPP. Combining dual and primal approaches prevents from increasing of computational time with contact area extension. See [8] for details.

2.2.4 Results

Let us examine how mentioned above modifications of G-I-P algorithm affect the computational speed. We consider test problem with 2,292 unknown variables. Let us examine how first three modifications speed up computations. In following plot (see Figure 5) the computation time is plotted along Y axis, X axis corresponds to number of fasteners installed between joined parts. Violet line corresponds to original algorithm and green line - to the version with all described modifications. It is clearly seen that computation time does increase if more forces are applied to riveted parts, but proposed modifications reduce the time more than twice in comparison with original version of algorithm.

The next bar chart in Figure 6 illustrates the computation time needed to solve test problem if 153 fasteners are installed using modified G-I-P algorithm. We consider several load cases with different initial approximations such as:

- Unconstrained minimum (blue column).
- Solution obtained for 93 fasteners installed (red column).
- Solution obtained for 113 fasteners installed (green column).
- Solution obtained for 133 fasteners installed (violet column).

According to the bar chart we may conclude that if we use obtained solution of previous problem we skip iterations that have already been made. Thus if external forces, for example, change from configuration of 133 to 153 fasteners we may reduce computational

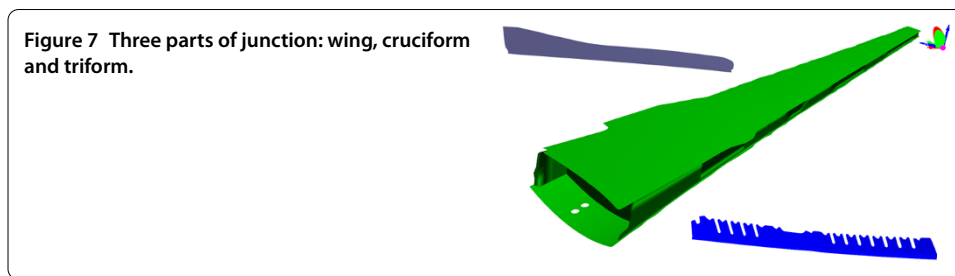
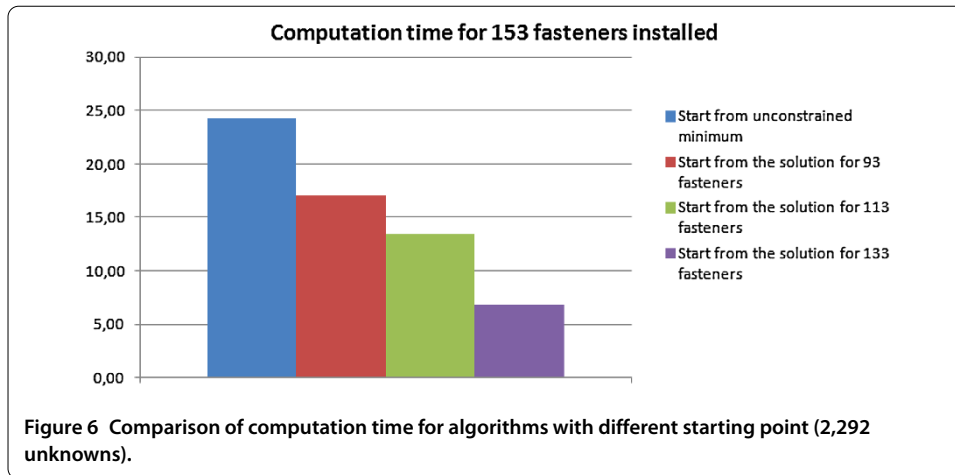


Figure 7 Three parts of junction: wing, cruciform and triform.

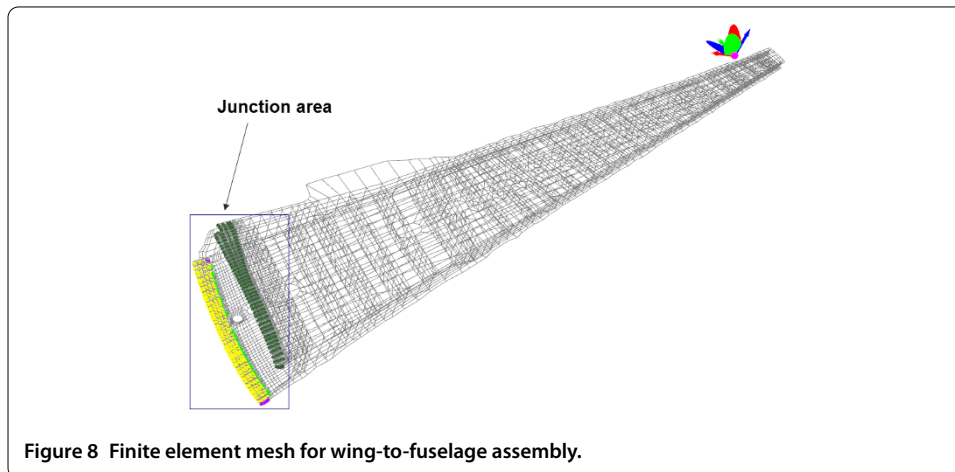


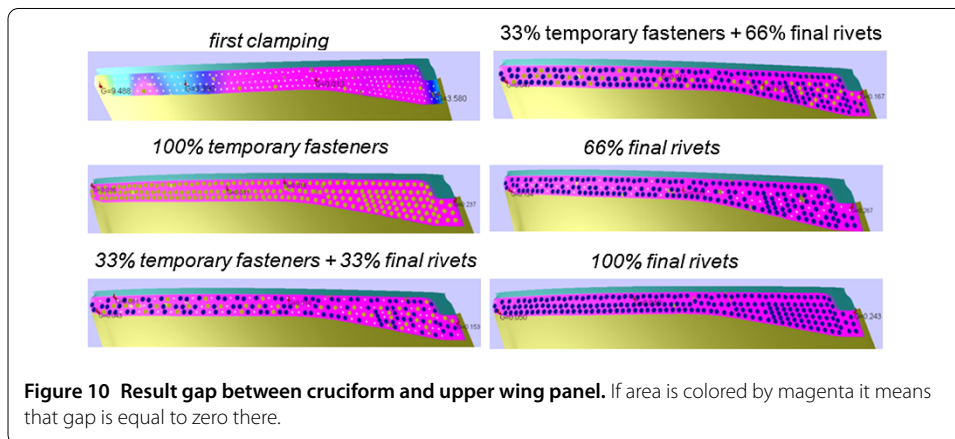
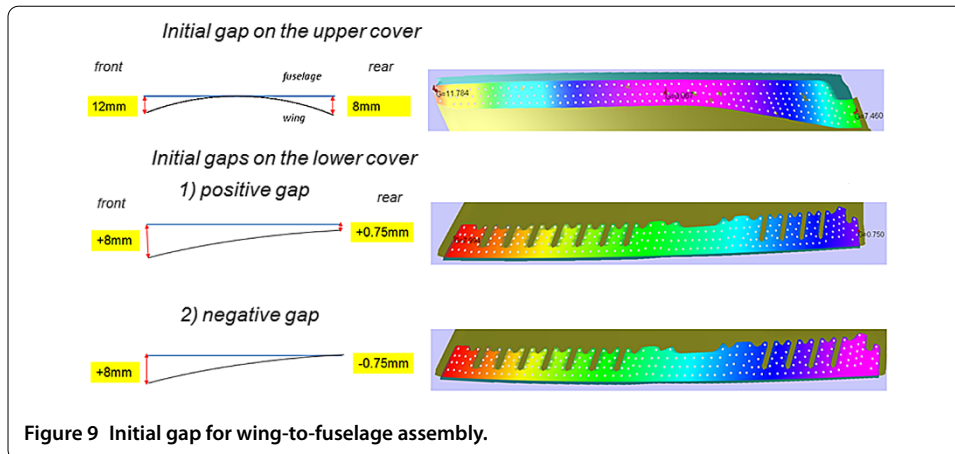
Figure 8 Finite element mesh for wing-to-fuselage assembly.

time from 24 to 6 seconds. This ‘warm start’ technology is quite profitable when making series of computations.

2.3 Applications

2.3.1 Wing-to-fuselage assembly simulation

First of all, we would like to consider the application of developed approach to the simulation of wing-to-fuselage assembly. Upper panel of wing is joined to cruciform, lower wing panel is joined to triform. All three parts are shown in Figure 7.



If we examine the finite element mesh of this model we may see that the junction area is quite small in comparison with the whole model (shown by colored dots in Figure 8). Thus this model perfectly fits the requirements of proposed methodology.

We are to set the initial gap (see Figure 9) and then observe the results when simulating different stages of assembly process.

The resulting gap fields are shown in Figures 10 and 11. The installed fasteners are shown by yellow dots, final rivets - by blue dots and free holes - by small grey dots.

2.3.2 Fuselage-to-fuselage assembly simulation

Also we present an example of fuselage-to-fuselage assembly simulation.

Let us consider two fuselage panels, as it is shown in Figure 12.

Figure 13 gives the finite element mesh adjacent to the junction area that is shown by green.

An example of simulation results for this case is given by Figure 14. Three installed fasteners are shown by yellow dots and free holes - by small grey dots. In the fasteners' neighborhood the gap is closed (shown by blue color).

The physical experiment on considered fuselage-to-fuselage assembly was made for verification of computation results. Figure 15 gives measured and computed gaps between riveted fuselage panels on the edge of junction area. The gap was measured in 42 points.

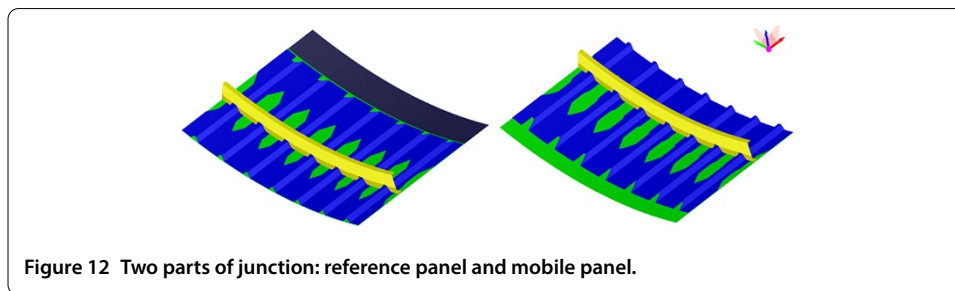
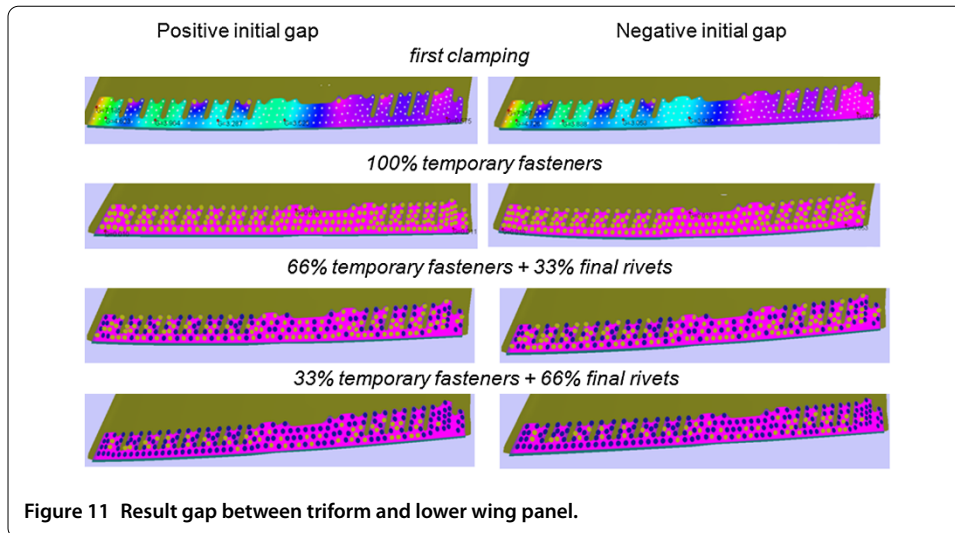


Figure 12 Two parts of junction: reference panel and mobile panel.

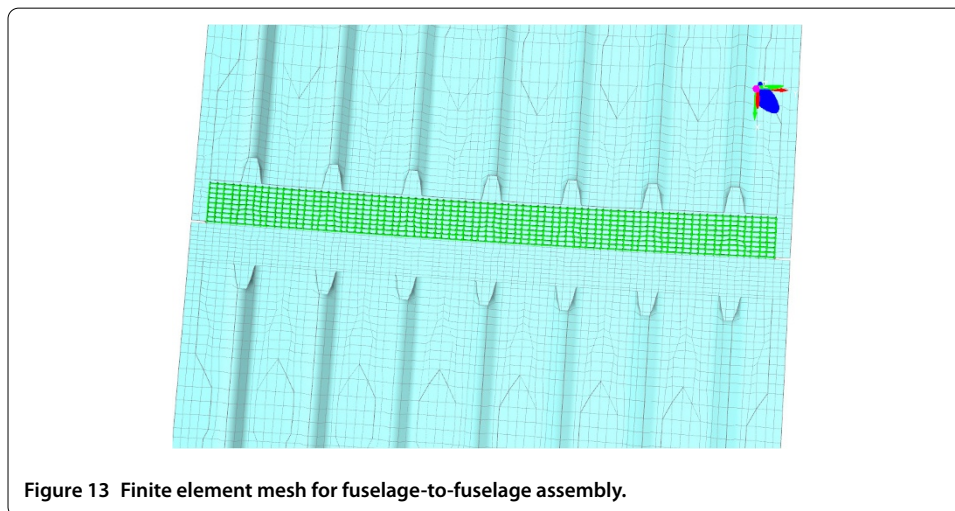


Figure 13 Finite element mesh for fuselage-to-fuselage assembly.

3 Conclusions

Developed approach is the most efficient for simulation of aircraft parts' riveting. The need to perform series of similar computations and a priori known narrow junction area are the main features that allow reducing computational time and resources.

The collaboration between St. Petersburg Polytechnic University and Airbus Company in the area of airframe assembly process simulation continues since 2006. Recently ASRP complex was entitled TRL 5 level by Airbus Company. TRL is the abbreviation for Tech-

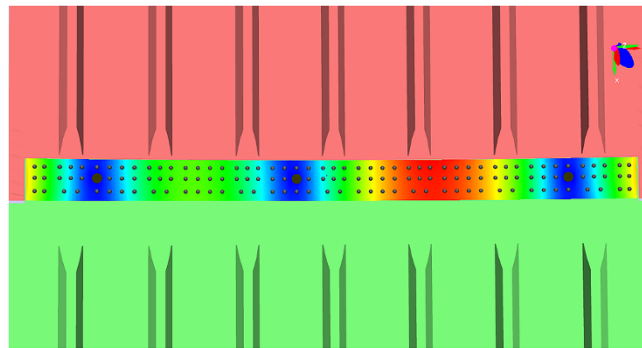


Figure 14 Computed displacements in fuselage-to-fuselage assembly. Rainbow coloring is used: minimal gap values are shown by blue color, maximal values are shown by red color.

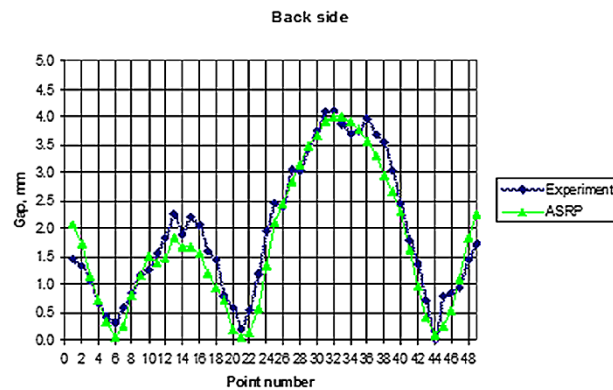


Figure 15 Measured and calculated gaps with three installed fasteners.

nology Readiness Level. TRL 5 means that the technology has passed the thorough testing of prototyping in representative environment. Basic technology elements integrated with reasonably realistic supporting elements. Prototyping implementations conform to target environment and interfaces (see http://esto.nasa.gov/files/trl_definitions.pdf).

Described methodology was fruitfully applied to simulation of real airframe junctions. In particular ASRP complex was successfully tested on airframe junctions for latest Airbus aircraft models like A350 and A320 NEO.

Competing interests

Our research was held out as a part of joint project between AIRBUS company and Saint Petersburg State Polytechnic University.

Authors' contributions

All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Author details

¹Department of Applied Mathematics, Saint Petersburg State Polytechnic University, Saint Petersburg, Russian Federation.

²Airbus, Toulouse, France.

Acknowledgements

We would like to thank Jacques Bouriquet, Nicolas Prioul, Claude Gimenez (AIRBUS) and professor Eugene Victorov (Saint Petersburg State Polytechnic University) for their invaluable assistance.

Received: 10 January 2014 Accepted: 25 March 2014 Published: 03 Jun 2014

References

1. Lupuleac S, Kovtun M, Rodionova O, Marguet B: **Assembly simulation of riveting process.** *SAE Int J Aerosp* 2010, **2**:193-198.
2. Lupuleac S, Rodionova O, Smirnov A, Shubnikov V, Bretagnol B: **Software complex for riveting process simulation.** *SAE Technical Paper* 2011, 2011-01-2772
3. Lupuleac S, Petukhova M, Shinder Y, Bretagnol B: **Methodology for solving contact problem during riveting process.** *SAE Int J Aerosp* 2011, **4**:952-957.
4. Pedersen P: **A direct analysis of elastic contact using super elements.** *Comput Mech* 2006, **37**:221-231.
5. Wriggers P: *Computational Contact Mechanics.* 2nd edition. Berlin: Springer; 2006.
6. Goldfarb D, Idnani A: **A numerically stable dual method for solving strictly quadratic programs.** *Math Program* 1983, **27**:1-33.
7. Powell MJD: **On the quadratic programming algorithm of Goldfarb and Idnani.** *Math Program Stud* 1985, **25**:46-61.
8. Lupuleac S, Shinder Y, Petukhova M, Yakunin S, Smirnov A, Bondarenko D: **Development of numerical methods for simulation of airframe assembly process.** *SAE Int J Aerosp* 2013, **6**:101-105.

10.1186/2190-5983-4-8

Cite this article as: Petukhova et al.: Numerical approach for airframe assembly simulation. *Journal of Mathematics in Industry* 2014, **4**:8

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com
