ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

FAST KINEMATIC LIMIT ANALYSIS OF FRP REINFORCED MASONRY VAULTS THROUGH A NEW GENETIC ALGORITHM NURBS-BASED APPROACH

Andrea Chiozzi[1], **Gabriele Milani**[2], **and Antonio Tralli**[1]

[1] Department of Engineering, University of Ferrara
1, Via Saragat, I-44122, Ferrara, Italy
e-mail: {andrea.chiozzi, tra}@unife.it

[2] Department of Architecture, Built Environment and Construction Engineering (A.B.C.)
Technical University of Milan,
31, Via Ponzio, I-20133, Milan, Italy
e-mail: gabriele.milani@polimi.it

Keywords: Masonry vaults, Limit analysis, NURBS, Genetic Algorithm.

Abstract. *In the present contribution, a novel Genetic Algorithm NURBS-based approach for the limit analysis of FRP reinforced masonry vaults based on an upper bound formulation is developed.Vaults geometry can be described by a NURBS representation of their mid-surface, which can be generated within any commercial free form modeler, together with information about the local thickness at each point of the surface. By exploiting the properties of NURBS functions, a mesh of the given surface, which still provides an accurate representation of the vaulted surface, can be obtained. Each element of the mesh is a NURBS surface itself and is idealized as a rigid body. Starting from the obtained rigid bodies assembly, an upper bound limit analysis problem with very few optimization variables can be devised and in which dissipation is allowed along element edges only. A possible dissipation at the interfaces between FRP and masonry is also considered in order to take into account, in an approximate but effective way, the possible delamination of the strips from the supports. Due to the very limited number of rigid elements used, the quality of the collapse load so found depends on the shape and position of the interfaces, where dissipation is allowed. Mesh adjustments are therefore needed which is carried out by adopting a simple meta-heuristic (like a standard Genetic Algorithm GA) approach of mesh adjustment. The strength of the proposed GA-NURBS method lies in the fact that even by using a mesh made of very few elements (which therefore require a negligible computational time to have an estimate of collapse loads), it is possible to obtain accurate load multipliers and failure mechanisms, thus exhibiting an edge over existing methods for the collapse analysis of masonry vaults in terms of computational efficiency.*

1 INTRODUCTION

Masonry vaults represent one of the most widespread structural typologies in the historical buildings of both Eastern and Western architecture. Therefore, the interest for their preservation is growing over time along with the need for developing new efficient tools to analyze and evaluate their load-bearing capacity.

As pointed out in [1, 2], it can be affirmed that the modern theory of limit analysis of masonry structures, which has been developed mainly by Heyman [3], is the most reliable tool to assess the ultimate load bearing capacity of masonry vaults. According to Heyman [3], limit theorems of plasticity, i.e. static (lower bound) theorem and kinematic (upper bound) theorem, can be applied to masonry structures provided that the following conditions are verified: i) the compressive strength of the material is infinite; ii) sliding between parts is prevented; iii) tensile strength of masonry is negligible.

Let us observe that for structures made of clay bricks and mortar, collapse generally occurs at small overall displacements. Moreover, in some cases sliding is possible though with a relatively high friction coefficient [4] and shear failure at the joints can be treated within the framework of non-associate plasticity [5]. Finally, although clay bricks masonry exhibits an almost zero tensile strength and a good compressive strength, the infinite compressive strength hypothesis is questionable and, as shown in [3], it is possible to include finite compressive strength within a limit analysis formulation. Nevertheless, material crushing play a minor role in the collapse behavior of masonry structures, except for very shallow segmental arches, pillars, towers and massive vertical structures.

Limit analysis can be also extended to the case of FRP (fibro-reinforced polymers) reinforced masonry structures [6].

Other essential aspects concerning actual masonry vaults should be considered, such as the effects due to material heterogeneity, the importance of the overall geometry for achieving the equilibrium, the importance of properly taking into account the infill and the presence of existing cracks [7].

The recently developed computational methods for masonry vaults, simple or reinforced, can be classified into two broad categories: the Finite Element methods developed both for nonlinear incremental analysis [8] and for limit analysis [9], and the thrust network methods [10-11] directly based on a lower bound formulation [12]. Practical application of these methods requires skilled users and, in the case of thrust network methods, the definition of an equilibrium surface for the vault, which is a priori unknown.

The present paper proposes a new NURBS-based approach [13] for the limit analysis of masonry vaults based on an upper bound formulation also allowing for the presence of FRP reinforcements, developing an idea proposed in [14] for masonry arches. NURBS (i.e. Non-Rational Uniform Bi-Spline) are special approximating base functions widely used in the field of 3D modeling [15]. A given masonry vault geometry can be represented by a NURBS parametric surface, which can be generated within any commercial free form modeler. By exploiting the properties of NURBS functions, a mesh of the given surface, which still provides an exact representation of the vaulted surface, can be obtained. Each element of the mesh is a NURBS surface itself and can be idealized as a rigid body.

Starting from the obtained rigid bodies assembly, an upper bound limit analysis formulation can be devised, which takes into account the main aspects of masonry material (i.e. negligible tensile strength and good compressive strength) and in which dissipation is allowed along element edges only. Furthermore, a meta-heuristic approach based on the implementation of a Genetic Algorithm allows adjusting the initial NURBS mesh until a good estimate of the collapse load multiplier is obtained. This happens when element edges describes the actual

failure mechanism. The strength of the proposed method lies in the fact that even by using a mesh made of very few elements, it is possible to obtain an accurate estimate of the load multiplier, thus exhibiting an edge over existing methods for the collapse analysis of masonry vaults in terms of computational efficiency,

The paper is organized as follows: in Section 2 a synthetic survey is given about how the geometric shape of a masonry vault can be described by a NURBS surface representation and a NURBS mesh can be defined on it. In Section 3, an upper bound limit analysis formulation is proposed, based on the NURBS geometric representation of the masonry vault, which allows to compute the collapse load for a set of given failure mechanisms. Section 4 outlines a Genetic Algorithm, which is capable of selecting the correct failure mechanism, by adequately adjusting the initial mesh. Finally, Section 5 is devoted to presenting several simple numerical results obtained by the proposed procedure.

2 NURBS GEOMETRIC DESCRIPTION

Description and computation of geometries in commercial CAD packages are based on B-Splines and NURBS approximating functions. More precisely, NURBS basis functions are built on B-splines basis functions, which are piecewise polynomial functions defined by a sequence of coordinates $\Xi = \{\xi_1, \xi_2, ..., \xi_{n+p+1}\}$, also known as the knot vector, where the so-called knots, $\xi_i \in [0,1]$, are points in a parametric domain, in which *p* and *n* denote the polynomial order and the total number of basis functions, respectively. Once the order of the basis function and the knot vector are known, the *i*-th B-spline basis function, $N_{i,p}$, can be computed by means of the Cox-de Boor recursion formula [15]. Given a set of weights, $w_i \in \mathbb{R}$, the NURBS basis functions, $R_{i,p}$, read

$$R_{i,p}(\xi) = \frac{N_{i,p}(\xi) w_i}{\sum_{i=1}^{n} N_{i,p}(\xi) w_i}. \tag{1}$$

NURBS share many properties with B-spline basis functions [13]. Among these, they are all nonnegative, they have a compact support, and build a partition of unity (PoU), that is

$$\sum_{i=1}^{n} N_{i,p}(\xi) = \sum_{i=1}^{n} R_{i,p}(\xi) = 1 \tag{2}$$

for each $\xi \in [0,1]$ [13]. Hence, according to Eqs. (1) and (2) B-spline basis functions can be thought of as NURBS basis functions when all weights w_i are equal to one. However, NURBS basis functions have the great advantage of representing exactly the geometry of a wide set of curves such as circles, ellipses, and parabolas [15], and of the surfaces that can be generated by these curves. Geometries that can be generated with B-spline and NURBS are obtained as linear combinations of basis functions [13]. If one considers a set of NURBS basis functions $R_{i,p}$, a NURBS curve of degree *p* is a parametric curve in the three-dimensional Euclidean space defined as

$$\mathbf{C}(u) = \sum_{i=1}^{n} R_{i,p}(\xi) \mathbf{B}_i \tag{3}$$

where coefficients $\mathbf{B}_i \in \mathbb{R}^3$ are known as control points. Unlike standard Lagrange and Hermite approximations, NURBS geometries do not usually interpolate these points. The conti-

nuity of the curve follows from that of the adopted basis functions [15], which is generally C^{p-1} throughout the domain. However, if a knot has multiplicity, *m*, the continuity decreases *m* times at that point [15]. Analogously, a NURBS surface of degree *p* in the *u*-direction and *q* in the *v*-direction is a parametric surface in the three-dimensional Euclidean space defined as

$$\mathbf{S}(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{m}R_{i,j}(u,v)\mathbf{B}_{i,j} \tag{4}$$

where $\{\mathbf{B}_{ij}\}$ form a bidirectional net of control points. A set of weights $\{w_{i,j}\}$ and two separate knot vectors in both *u* and *v* directions must be defined. Given a NURBS surface $\mathbf{S}(u,v)$, isoparametric curves on the surface can be defined by fixing one parameter in the parameter space and letting the other vary. By fixing $u=u_0$ the isoparametric curve $\mathbf{S}(u_0,v)$ is defined on the surface $\mathbf{S}$, whereas by fixing $v=v_0$ the isoparametric curve $\mathbf{S}(u,v_0)$ is obtained.

Many commercial free form surface modelers, such as Rhinoceros® [16], utilize NURBS representation and its properties to generate and manipulate surfaces in the three-dimensional space. In what follows, simple vault geometries have been generated within Rhinoceros and the resulting NURBS structure have been imported within a MATLAB® environment through the IGES (Initial Graphics Exchange Specification) standard [17]. Once the NURBS structure has been transferred to the MATLAB® environment, it is possible to manipulate it by exploiting NURBS properties in order to define a NURBS mesh on the given surface, i.e. a mesh in which each element is a NURBS surface itself. When working with simple surfaces like the one considered in the present contribution, the easiest way to generate a NURBS mesh on the given surface is to define a subdivision of the two-dimensional parameters space *u*-*v*, which follows from subdividing the knot vectors in both *u* and *v* directions into equal intervals. The resulting mesh is formed by isoparametric curves on the surface in the three-dimensional Euclidean space. Each element is a NURBS surface and its edges are branches of isoparametric curves belonging to the initial surface. More precisely, the counter-image of each element of the mesh is a rectangle $S_{ij}=[u_i,u_{i+1}]\times[v_j,v_{j+1}]\in\mathbb{R}^2$ defined in the parameters space. For each element (which can be denoted by E_{ij}), the area of the surface can be computed through the following relation:

$$A_{ij}=\int_{S_{ij}}dS=\int_{u_i}^{u_{i+1}}\int_{v_j}^{v_{j+1}}\left\|\mathbf{S}_u\times\mathbf{S}_v\right\|du\,dv \tag{5}$$

where $\mathbf{S}_u$ and $\mathbf{S}_v$ are partial derivatives of the parametric surface $\mathbf{S}(u,v)$ in the *u* and *v* directions. Analogously, the center of mass of each element may be computed with the following relation:

$$\mathbf{c}=\frac{1}{A_{ij}}\int_{S_{ij}}\mathbf{x}\,dS=\frac{1}{A_{ij}}\int_{u_i}^{u_{i+1}}\int_{v_j}^{v_{j+1}}\mathbf{S}(u,v)\left\|\mathbf{S}_u\times\mathbf{S}_v\right\|du\,dv \tag{6}$$

Computations can be numerically carried out using a standard Gauss quadrature method.

3 KINEMATIC LIMIT ANALYSIS

Limit analysis is a powerful tool to assess the structural safety level of a masonry construction. As already discussed, given the NURBS geometric representation of the vaulted surface, a NURBS mesh can be defined on the same surface. Each element of the mesh, which is a NURBS surface itself, can be regarded as a rigid body. Starting from the geometrical proper-

ties of each element, an upper bound formulation can be outlined and implemented through a linear programming algorithm in order to assess the ultimate load bearing capacity of a given masonry vault. This paragraph summarizes the proposed upper bound formulation.

Be N_E the number of elements composing the NURBS mesh, which geometrically represents the vaulted surface. Each element is considered as a rigid element. Thus, the kinematics of each element is determined by the six (three translational and three rotational) generalized velocity components $\{u_x^i, u_y^i, u_z^i, \Phi_x^i, \Phi_y^i, \Phi_z^i\}$ of its center of mass G_i, expressed in a global reference system $Oxyz$. On the structure, dead loads $\mathbf{F}_0$ and live loads $\mathbf{\Gamma}$ are acting. Internal dissipation is assumed to occur only along element interfaces. Indicating by N_I the number of interfaces, total internal dissipation power D_{int} is equal to the sum of the power dissipated along each interface P_{int}^i, which is defined in Section . Furthermore, total internal dissipation power D_{int} is equal to the sum of the powers of live ($\mathbf{1} \cdot \Gamma$) and dead ($\mathbf{F_0}$) loads, indicated as $P_{\mathbf{\Gamma}}$ and $P_{\mathbf{F_0}}$ respectively:

$$D_{\text{int}} = \sum_{i=1}^{N_I} P_{\text{int}}^i = P_{\mathbf{\Gamma}} + P_{\mathbf{F_0}} \tag{7}$$

Γ is a load multiplier. The linear programming problem related to the kinematic formulation of limit analysis consists in an appropriate minimization of the load multiplier Γ under the action of suitable constraints, which are described in the following Subsections. The vector of unknowns of the linear programming problem, $\mathbf{X}$, contains the six generalized velocity components for each element and a number of plastic multipliers along each interface which will be defined in Subsection 3.2.

3.1 Geometric constraints

Vertex belonging to element free edges, which do not constitute an element interface, can be subjected to external kinematic constraints, by imposing an assigned value for translational and/or rotational velocities at these points. For each of such vertex V_j, kinematic constraints can be expressed in terms of generalized velocities of the center of mass of the *i*-th element they belong. For example, in case only translational velocities of a given vertex V_j, belonging to element *i*, are constrained to zero, the following relation holds as a geometric constraint:

$$\mathbf{u}_{V_j} = \mathbf{u}^i + \mathbf{R}\left[\mathbf{x}_{V_j} - \mathbf{x}_{G_i}\right] = \mathbf{0} \tag{8}$$

where $\mathbf{u}_{V_j} = [u_x^{V_j}, u_y^{V_j}, u_z^{V_j}]^T$ are the three translational velocity components of the vertex V_j, $\mathbf{u}^i = [u_x^i, u_y^i, u_z^i]^T$ are the three (unknown) translational velocity components of the center of mass of element *i* to whom vertex V_j belongs, and $\mathbf{R}$ is a rotation matrix whose elements are the (unknown) generalized rotational velocities of the center of mass of element *i*. In general, all linear geometric constraints can be re-written in the following standard form:

$$\mathbf{A}_{eq,geom}\mathbf{X} = \mathbf{b}_{eq,geom} \tag{9}$$

where $\mathbf{A}_{eq,geom}$ is the matrix of geometric constraints and $\mathbf{b}_{eq,geom}$ the corresponding vector of coefficients.

3.2 Compatibility constraints

In order to enforce plastic compatibility along interfaces and correctly evaluate dissipation power, intrados and extrados edges of each interface have been subdivided into an assigned number $(N_{sd}+1)$ of points P_i . On each point P_i, a local reference system $(\mathbf{n},\mathbf{s},\mathbf{t})$ have been defined, where $\mathbf{n}$ is the unit vector normal to the interface, $\mathbf{s}$ is the tangential unit vector in the longitudinal direction and $\mathbf{t}$ is the tangential unit vector in the transversal direction.

On each point P_i of each interface, which separates the two elements E' and E'', the following compatibility equation must hold:

$$\Delta\tilde{\mathbf{u}} = \dot{\boldsymbol{\lambda}}\frac{\partial f}{\partial \boldsymbol{\sigma}} \tag{10}$$

where $\boldsymbol{\sigma}=[\sigma_{nn},\sigma_{ns},\sigma_{nt}]$ is the stress vector acting on P_i in the three local reference directions, $f(\boldsymbol{\sigma})$ is a suitable yield function and $\dot{\boldsymbol{\lambda}}$ is an unknown plastic multiplier vector. In Eq. (10), $\Delta\tilde{\mathbf{u}}$ is the representation in the local reference system of the quantity $\Delta\mathbf{u}$ in the global reference system which is defined as:

$$\Delta\mathbf{u} = \mathbf{u}'_{P_i} - \mathbf{u}''_{P_i} \tag{11}$$

where $\mathbf{u}'_{P_i}$ is the vector composed by the three translational velocity components of the point P_i seen as belonging to element E' and $\mathbf{u}''_{P_i}$. $\Delta\mathbf{u}$ is related to $\Delta\tilde{\mathbf{u}}$ through the following relation:

$$\Delta\tilde{\mathbf{u}} = \tilde{\mathbf{R}}\Delta\mathbf{u} \tag{12}$$

where $\tilde{\mathbf{R}}$ is a suitable 3×3 rotation matrix whose rows are respectively the components of the three local vectors $(\mathbf{n},\mathbf{s},\mathbf{t})$ expressed in the global reference system.

For the sake of simplicity, the yield surface $f(\boldsymbol{\sigma})$ have been defined in the stress-space as the parallelepiped defined by the inequalities $-f_c \le \sigma_{nn} \le f_t$, $-\tau_{0,ns} \le \sigma_{ns} \le \tau_{0,ns}$, $-\tau_{0,nt} \le \sigma_{nt} \le \tau_{0,nt}$ where f_c, f_t, $\tau_{0,ns}$, $\tau_{0,nt}$ are material parameters corresponding respectively to compression strength, tensile strength, shear strength in the direction tangential to the interface midline and shear strength in the direction orthogonal to the interface midline. If the interface is reinforced by FRP strips at intrados or extrados f_t is replaced by the delamination stress f_d computed according to [18] in order to take into account dissipation due to FRP delamination.

With the assigned yield surface, Eq. (10) simplifies and must hold for each point P_i of each interface. On each point P_i, six unknown plastic multipliers $\dot{\lambda}^+_{nn},\dot{\lambda}^-_{nn},\dot{\lambda}^+_{ns},\dot{\lambda}^-_{ns},\dot{\lambda}^+_{nt},\dot{\lambda}^-_{nt}$ are defined. Therefore, the total number of unknown plastic multipliers is equal to $6\cdot(N_{sd}+1)\cdot 2N_I$.

3.3 Non-negativity of plastic multipliers

An additional constraint which must be included into the linear programming problem is the non-negativity of each plastic multiplier:

$$\dot{\lambda}_{ij} \geq 0. \tag{13}$$

3.4 Normality condition

The last condition to be applied is the so-called normality condition which requires that the external power dissipated by the live load $\mathbf{1} \cdot \Gamma$ set equal to one, is itself equal to one, i.e.:

$$P_{\Gamma=1} = 1 \tag{14}$$

This condition allows to rewrite Eq. (7) in the following way:

$$\Gamma = \sum_{i=1}^{N_I} P_{\text{int}}^i - P_{F_0} \tag{15}$$

3.5 Internal dissipated power and linear programming problem

On each interface i , covering the surface S_i, the internal dissipated power is defined as the integral:

$$P_{\text{int}}^i = \int_{S_i} \boldsymbol{\sigma} \cdot \Delta\tilde{\mathbf{u}} \, dS \tag{16}$$

in the local reference system, where both $\boldsymbol{\sigma}$ and $\Delta\tilde{\mathbf{u}}$ have been defined in Subsection 3.2. Therefore, remembering Eq. (15) and following the kinematic theorem of limit analysis, the related linear programming problem can be stated as follows:

$$\min \left\{ \sum_{i=1}^{N_I} P_{\text{int}}^i - P_{\mathbf{F_0}} \right\} \tag{17}$$

under geometric constraints (9), compatibility constraints (10), non-negativity of plastic multipliers constraints (13) and the normality condition (14). The unknowns of the linear programming problem are the $6 \cdot N_E$ generalized velocity components of the center of mass of each element and the $6 \cdot (N_{sd} + 1) \cdot 2N_I$ plastic multipliers at each point of each interface.

4 GENETIC ALGORITHM

A genetic algorithm is used to progressively modify the mesh in order to find the minimum collapse multiplier among all possible configurations and therefore to determine the actual collapse mechanism. A genetic algorithm is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. The algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm randomly selects individuals from the current population and uses them as parents to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution.

A NURBS mesh of a vaulted surface, is determined by a given number N_{par} of real parameters $p_1, p_2, ..., p_{Npar}$, that depend on the type of collapse mechanism which must be detected. A given NURBS mesh is regarded as an individual and each individual (or chromosome), is written as an array with $1 \times N_{par}$ elements:

$$individual = [p_1, p_2, ..., p_{Npar}] \tag{18}$$

Each individual has a cost, found by evaluating the cost function *f* at the parameters $p_1, p_2, ..., p_{Npar}$. The cost function *f* is defined as a function which outputs the collapse load multiplier λ_c for every assigned individual (i.e. an assigned mesh on the surface) through the implementation of the limit analysis procedure described in Section 3:

$$\lambda_c = f(individual) = f(p_1, p_2, ..., p_{Npar}) \tag{19}$$

To begin the genetic algorithm, we define an initial population of N_{ipop} individuals. A matrix represents the population with each row in the matrix being a $1 \times N_{par}$ array (individual) of continuous parameters values. Given an initial population of N_{ipop} individuals, the full matrix of $N_{ipop} \times N_{par}$ random values is generated by

$$IPOP = (hi - lo) \times \mathbf{random}\{N_{ipop}, N_{par}\} + lo \tag{20}$$

where $\mathbf{random}\{N_{ipop}, N_{par}\}$ is a function that generates an $N_{ipop} \times N_{par}$ matrix of uniform random numbers, *hi* and *lo* are the highest and lowest number in the parameter range.

Individuals are not all "create equal": each one's worth is assessed by the cost function. In order to decide which chromosomes in the initial population of individuals are fit enough to survive and reproduce offspring in the next generation the N_{ipop} costs and associated individuals are ranked from lowest cost to highest cost. We retain the best N_{pop} members of the population for the next iteration of the algorithm and the rest die off. This process is called natural selection and from this point on, the size of the population at each generation is N_{pop}. An equal number of mothers and fathers is selected within the N_{pop} individuals, which pair in some random fashion. There are various reasonable ways to pair individuals [19]. In this paper, a weighted cost selection with assigned probabilities is used [19]. Each pair produces two offspring that contain traits from each parent. Mating is carried out by choosing one or more points in the chromosome to mark as the crossover points and the, parameters between these points are merely swapped between the two parents.

Finally, if care is not taken, the genetic algorithm converges too quickly into one region of the cost surface and this may be not good if the problem we are modeling has several local minima, in which the solution may get trapped. To avoid this problem of overly fast convergence, we force the routine to explore other areas of the cost surface by randomly introducing changes, or mutations, in some of the parameters. A mutation rate of 15% is chosen.

5 NUMERICAL EXAMPLES

In this Section, two numerical examples of NURBS based kinematic limit analyses of both reinforced and unreinforced masonry vaults are synthetically described. For each example, the mid-surface of the vault have been modeled with the 3D free form modeler Rhinoceros® and the corresponding NURBS structure have been imported within a MATLAB® environment using the IGES protocol. The limit analysis procedure described in Section 3 have been implemented and the collapse mechanism is determined by suitably adjusting the mesh through the genetic algorithm described in Section 4.

5.1 Unreinforced skew barrel vault

The proposed GA-NURBS approach is applied to the skew circular arch experimentally tested in [20]. The arch, named *Skew 2* in [20], has a clear square span of 3000 mm, a rise of 750 mm and a skew of 45 degrees. The width of the barrel was approximately 670 mm and the average thickness 215 mm. The arch was constructed using Class A engineering bricks were on two reinforced concrete abutments representing rigid supports. The geometry of the arch is reported in Fig. In the test, a concentrated load *P* was applied under force control at the three quarter span mid-width of the arch barrel. The load was monotonically increased up to 17.4kN when collapse occurred because of the formation of cracks extending in the mortar joints through the whole width of the arch, giving rise to a 3D failure mode typical of skewed masonry arches. An average brickwork compression strength f_c of 2.4 MPa and a tensile strength f_t of 0.2 MPa were measured, whereas a shear strength τ of 0.1 MPa is assumed. Average specific weight of brickwork is $22\,kN/m^3$. The initial NURBS mesh of the vaulted surface is formed by three quadrangular elements. A single centered vertical live load of $\lambda \cdot 1kN$ is applied at 1/4L.

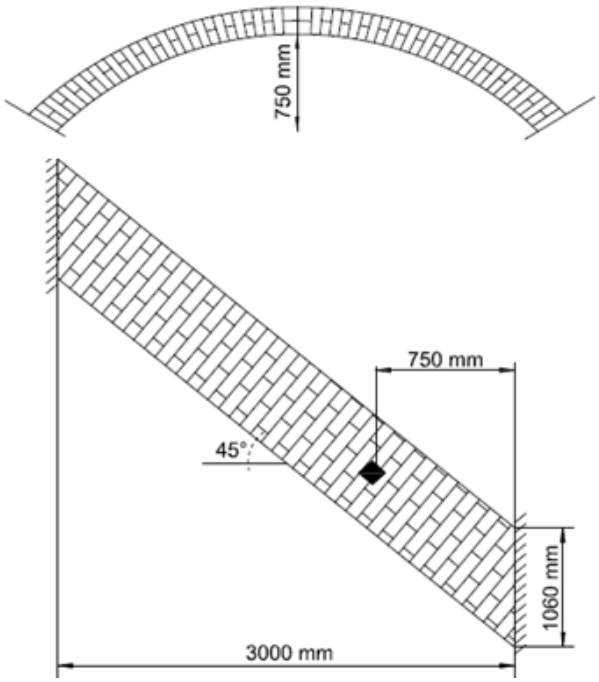

Figure 1: Skew arch geometry in the test configuration described in [20].

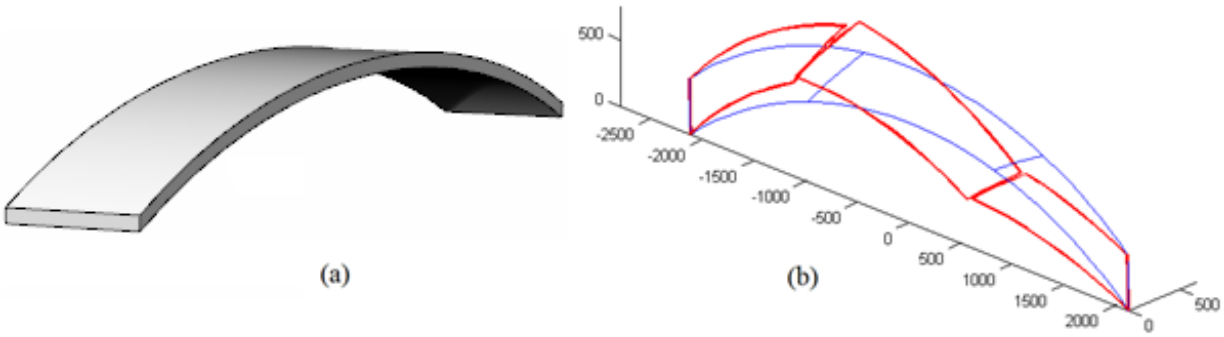

Figure 2: (a) 3D NURBS model of the skew arch experimentally tested in [20] generated with Rhinoceros®. (b) Three-element NURBS mesh (blue) and collapse mechanism from kinematic limit analysis (red).

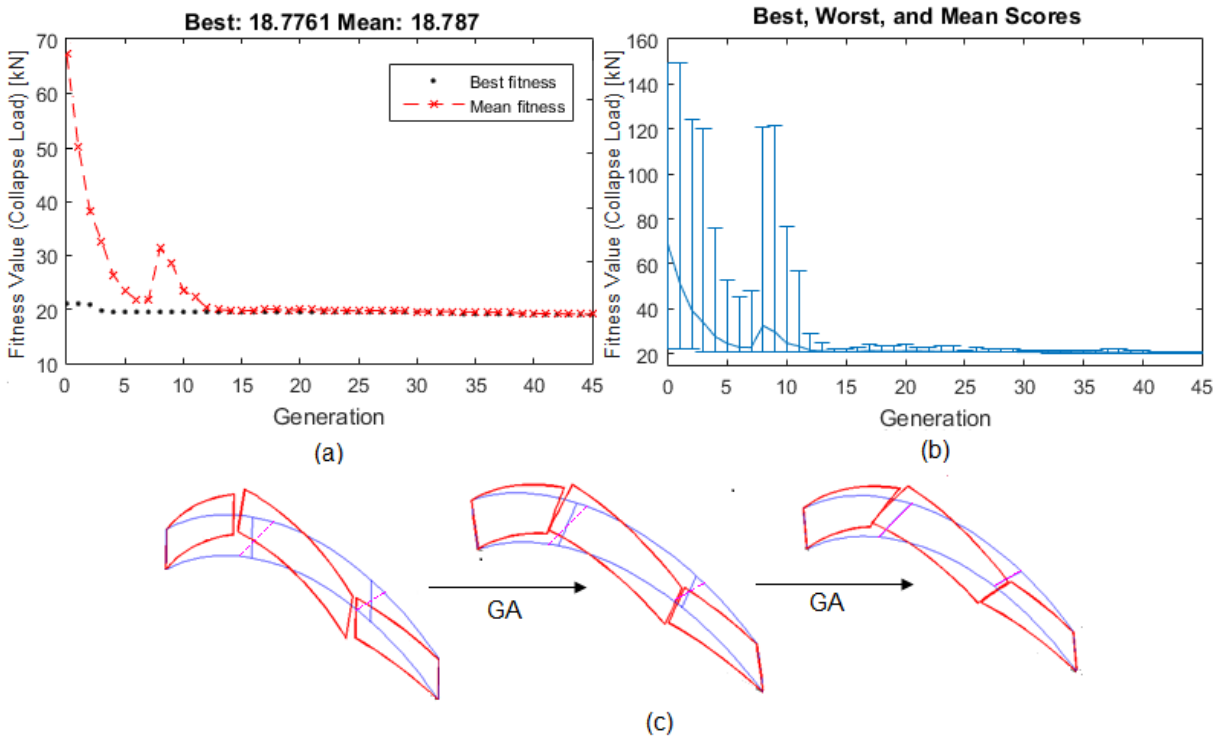

Figure 3: Skew arch: convergence of the genetic algorithm towards the optimal solution in terms of best fitness and mean value (a) and in terms of best, worst and mean scores (b) at each generation; evolution of the free interfaces towards the optimal solution (c).

The genetic algorithm allows evaluating the optimal position of the two active interfaces, in order to minimize the collapse load multiplier and therefore obtaining the actual collapse mechanism. Due to the point load presence, the position of the active interfaces is governed by three parameters: two parameters fix the extremes of the unloaded interface, whereas a third parameter fixes the position of the loaded interface (since this interface is bound to pass though the load application point). In the genetic algorithm an initial population of 40 individuals have been chosen, each individual being a 1x3 vector.

A collapse load multiplier $\lambda = 18.78$ have been obtained. Fig.2(a) shows the 3D NURBS model of the vault generated within Rhinoceros® and Fig.2(b) depicts the computed collapse mechanism, which proves to be equal to the one observed in [20].

As shown in Fig. 3(a-b), the algorithm presents a fast convergence towards the optimal solution and the final best fitness value is obtained after the first four generations. Fig. 3(c) represents the evolution of the mesh towards the optimal solution.

5.2 FRP reinforced square barrel vault

At first, a comparison with the experimental tests presented in [21] and later analyzed in [9, 14] for the unreinforced square barrel vault is carried out. In [21] the ultimate strength of a segmental masonry arch was tested, with a clear span of 3 m, an inner radius of 2.5 m and a sagitta of 0.5 m. The arch is a one-head brick structure with depth equal to 0.10 m and width equal to 1.25 m. The test-arch had 51 layers and was built with Rijswaard soft mud bricks and 1:2:9 mortar. Brick compressive strength was 27 MPa and mortar compressive strength was 2.5 MPa. The test-arch was loaded with four concentrated loads, applied by four hydraulic jacks 600 mm centre to centre. In Fig. 5a the geometry of the test-arch and its loading conditions are reported. Only the second concentrated load from the left was increased until failure, whereas the remaining loads were maintained constant at the values of 5.9, 9.1 and 9.1 kN

respectively. At failure, a four hinges collapse mechanism was observed in [21], which is depicted in Fig. 6a and measured a collapse load equal to 40.7 kN at the second jack. An average brickwork compression strength f_c of 2.4 MPa, a tensile strength f_t of 0.1 MPa and a shear strength τ of 0.1 MPa are assumed. Average specific weight of brickwork is $20\,kN/m^3$.

The described test-arch has been modeled within Rhino and its 3D model is reported in Fig. 5b. The initial NURBS mesh of the vaulted surface is formed by four quadrangular elements.

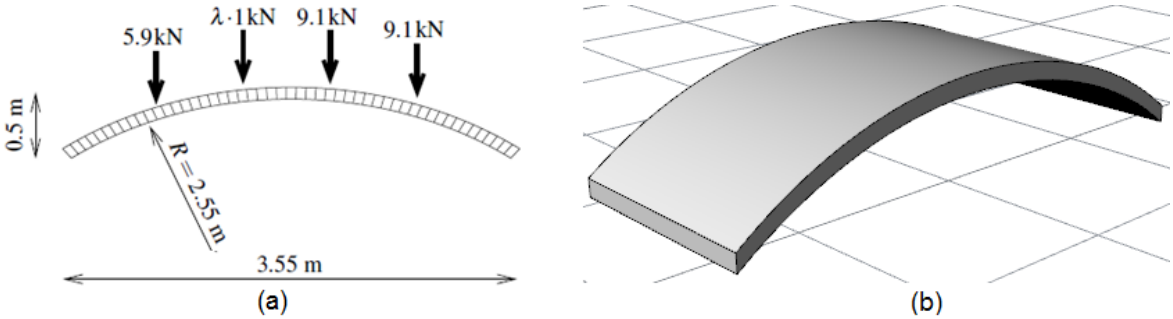

Figure 5: (a) Schematic representation of the segmental masonry test-arch used in [21]: geometry and loading conditions. (b) 3D NURBS model of the arch experimentally tested in [21] generated with Rhinoceros®.

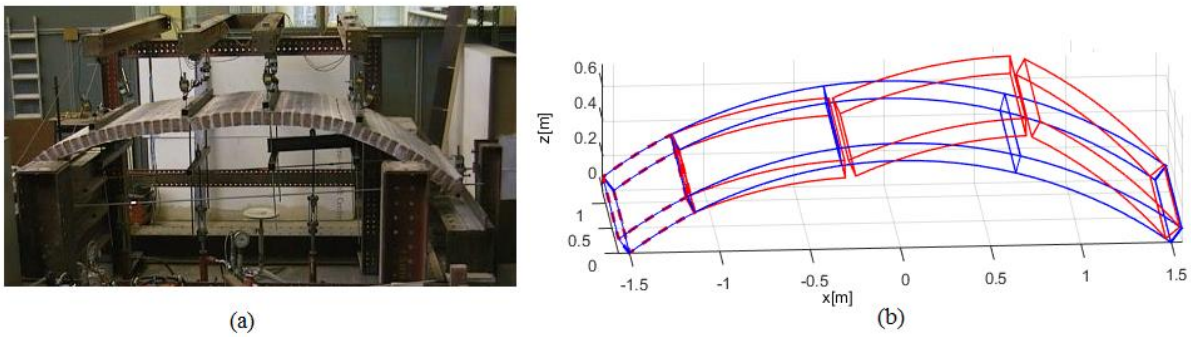

Figure 6: (a) Experimental failure mechanism obtained in [21]. (b) Failure mechanism computed through the GA-NURBS approach.

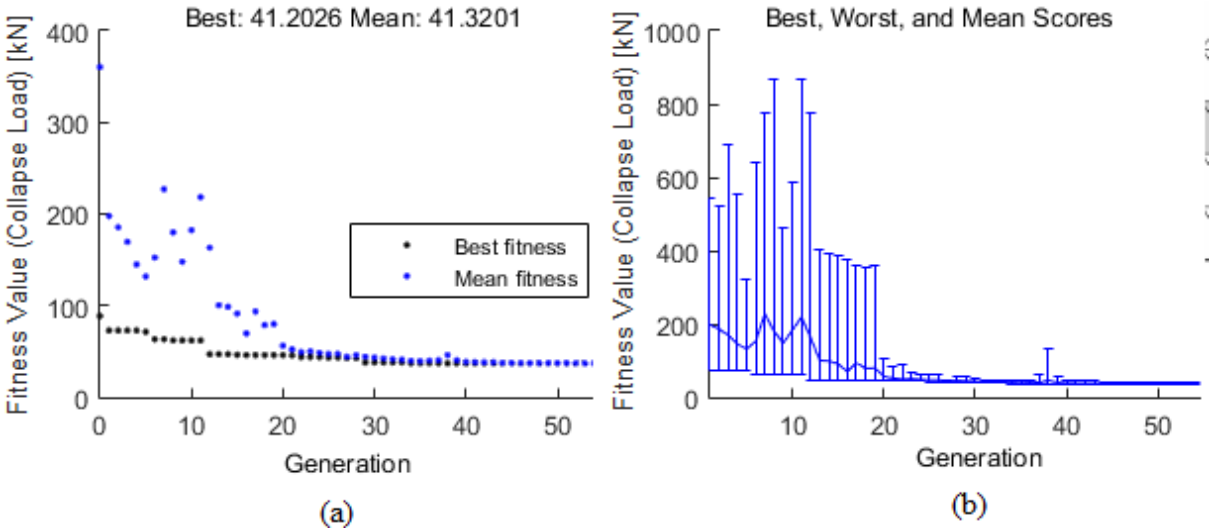

Figure 7: Square barrel vault: convergence of the genetic algorithm towards the optimal solution in terms of best fitness and mean value (a) and in terms of best, worst and mean scores (b) at each generation.

The genetic algorithm allows evaluating the optimal position of the active interfaces, in order to minimize the collapse load multiplier and therefore obtain the actual collapse mechanism. Only the second point load from the left have been marked as a live load. Collapse will occur after formation of four plastic hinges. Thus, the position of the active interfaces is governed by four parameters. In the genetic algorithm an initial population of 20 individuals have been chosen, each individual being a 1x4 vector. A collapse load multiplier $\lambda = 41.2$ have been obtained. Fig.6(b) depicts the computed collapse mechanism, which proves to be equal to the one observed in [21] and depicted in Fig. 6(a). As shown in Fig. 7, the algorithm presents a fast convergence towards the optimal solution.

Finally, the same vault was analyzed by assuming FRP reinforcement set on the extrados (Fig. 8). In order to prevent the formation of the plastic hinges indicated in Fig. 6(b), we suppose to strengthen the arch by means of two sets of FRP strips (width 100mm). The first set of strips is disposed at the extrados of the arch, whereas the second is applied at the intrados. The reinforcement tissue has thickness of 0.2 mm, Young's elastic modulus is assumed equal to 164 GPa (for tensile stress only) and an ultimate strain of 2% is adopted. FRP delamination stress f_d has been calculated by following the Italian FRP Design Guidelines [18]. Masonry strength parameters have been assumed the same of the unreinforced arch. The NURBS model is kept unchanged and so is the number of parameters governing the problem.

Again, the genetic algorithm allows evaluating the optimal position of the active interfaces, in order to minimize the collapse load multiplier and therefore obtaining the actual collapse mechanism.

The GA-NURBS approach allows to compute a collapse load multiplier $\lambda = 86.3$ of the FRP reinforced arch, which is in agreement with the results presented in [6].

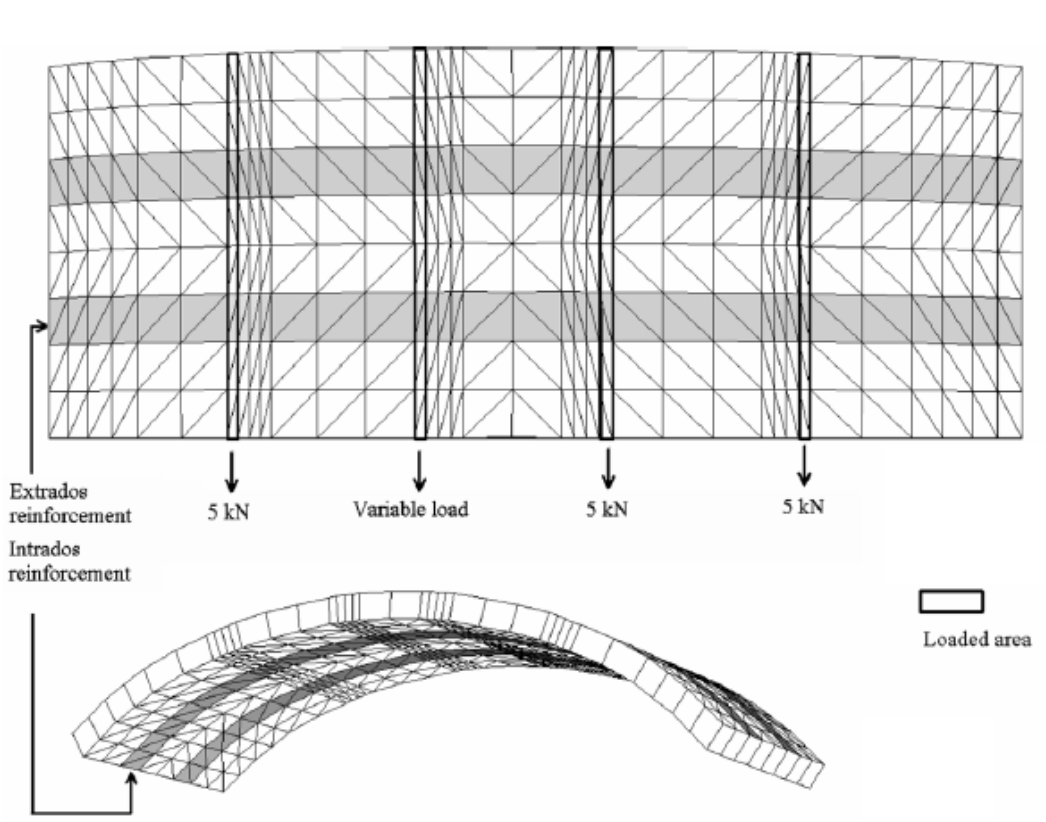

Figure 8: Reinforced parabolic arch. FRP strips dimensions and disposition.

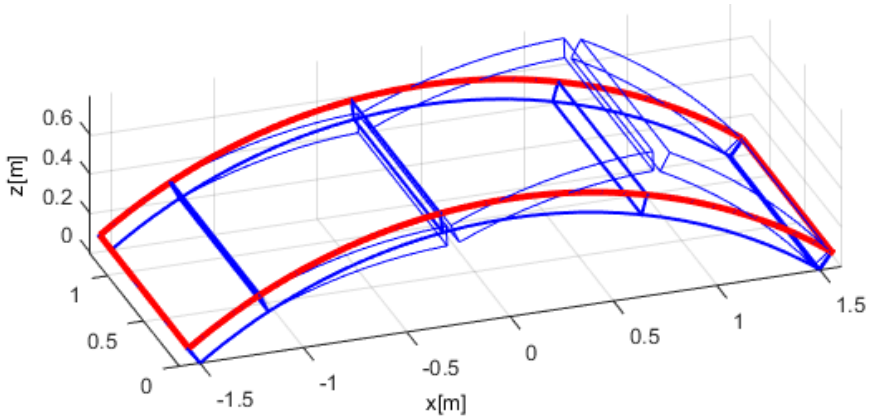

Figure 9: Square barrel vault: convergence of the genetic algorithm towards the optimal solution in terms of best fitness and mean value (a) and in terms of best, worst and mean scores (b).

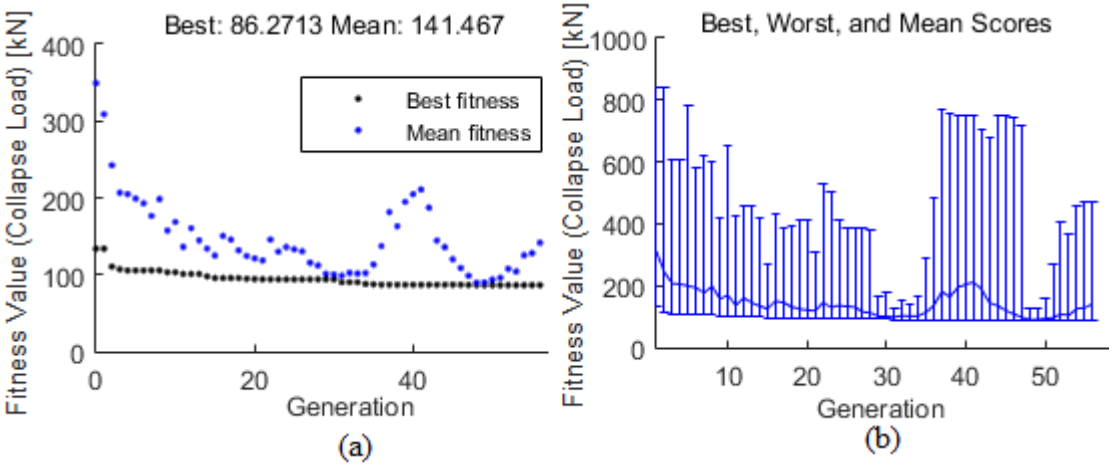

Figure 10: Square barrel vault: convergence of the genetic algorithm towards the optimal solution in terms of best fitness and mean value (a) and in terms of best, worst and mean scores (b).

In Figure 9 the four hinges collapse mechanism is shown: in this case the mechanism develops only after FRP delamination has occurred at the extrados hinges. The thick red line represents the outline of the FRP reinforcement.

As shown in Fig. 10, the algorithm presents a fast convergence towards the optimal solution and the final best fitness value is obtained after few generations.

REFERENCES

[1] M. Como, *Statics of historic masonry constructions,* Springer-Verlag, Berlin, 2013.

[2] S. Huerta, The analysis of masonry architecture: a historical approach. *Architectural Science Review*, **51**, 297-328, 2008.

[3] J. Heyman, *The stone skeleton: structural engineering of masonry architecture,* Cambridge University Press, UK, 1995.

[4] G. Vasconcelos, P.B. Lourenco, Assessment of the In-plane shear strength of stone masonry walls by simplified models. In *Proc. 5th International Conference of Structural Analysis of Historical Construction,* New Delhi, India, 2006.

[5] M. Gilbert, C. Casapulla, H.M. Ahmed, Limit analysis of masonry block structures with

non-associative frictional joints using linear programming. *Computers and Structures*, **84**, 873-887, (2006).

[6] G. Milani, E. Milani, A. Tralli, Upper bound limit analysis model for FRP–reinforced masonry curved structures. Part II: Structural analyses. *Computers and Structures*, **87**, 1534-1558, 2009.

[7] A. Tralli, C. Alessandri, G. Milani, Computational methods for masonry vaults: a review of recent results. *The Open Civil Engineering Journal*, **8**, 272-287, 2014.

[8] G. Milani, A. Tralli, A simple meso-macro model based on SQP for the non-linear analysis of masonry double curvature structures. *International Journal of Solids and Structures*, **49**, 808-834, 2012.

[9] G. Milani, E. Milani, A. Tralli, Limit analysis of masonry vaults by means of curved shell finite elements and homogenization. *International Journal of Solids and Structures*, **45**, 5258-5288, 2008.

[10] P. Block, T. Ciblac, J.A. Ochsendorf, Real-time limit analysis of vaulted masonry buildings. *Computers and Structures*, **84**, 1841-1852, 2006.

[11] P. Block, L. Lachauer, Three-dimensional (3D) equilibrium analysis of Gothic masonry vaults. *International Journal of Architectural Heritage*, **8**, 312-335, 2014.

[12] M. Angelillo, E. Babilio, A. Fortunato, Singular stress-fields for masonry-like vaults. *Continuum Mechanics and Thermodynamics*, **25**, 423-441, 2013.

[13] L. Piegl, W. Tiller, *The NURBS Book*, Springer, 1995.

[14] A. Chiozzi, M. Malagù, A. Tralli, A. Cazzani, ArchNURBS: NURBS-Based Tool for the Structural Safety Assessment of Masonry Arches in MATLAB. *ASCE Journal of Computing in Civil Engineering*, **30**, 2016.

[15] J. A. Cottrell, T.J.R. Hughes, Y. Bazilevs, *Isogeometric analysis: toward integration of CAD and FEA*, John Wiley & Sons, 2009.

[16] R. McNeel and Associates, *Rhinoceros, modeling tools for designers*, 5.0, www.rhino3D.com, 2015.

[17] US Product Data Association, *IGES 5.3 (ANSI-1996)*, 1996.

[18] National Research Council, *Instructions for the design, building and control of static retrofitting interventions through FRP composites*, DT200-2013, 2013.

[19] R.L. Haupt, S.E. Haupt, *Practical Genetic Algorithms*, John Wiley & Sons, 1998.

[20] J. Wang, C. Melbourne, The 3-Dimensional Behaviour of Skew Masonry Arches. *The British Masonry Society*, London, UK (1996).

[21] A. Vermeltfoort, Analysis and experiments of masonry arches. In: *Proceedings of Historical Constructions*, R.B. Lourenço and P. Roca Eds., Guimaraes, Portugal, (2001).