Fabrication of Accurate Bone Implant Geometry Using Puzzle Solving Technqiue

Rashmi Uddanwadikar
Assistant Professor, Dept of Mechanical Engg., VNIT Nagpur, India
rashmiu71@rediffmail.com

Vikram Uddanwadiker
TUV Nord, Mumbai, India
vikramu71@rediffmail.com

Shital Chiddarwar
Assistant Professor, Dept of Mechanical Engg., VNIT Nagpur, India
s.chiddarwar@gmail.com

***Abstract*— In the cases related to comminuted fractures, a part of bone is damaged/ crushed (comminuted) to the point of being missing altogether. As an example this type of trauma could occur from military injuries like gunshot wounds, explosives, motor vehicle accidents, or falling from excessive heights; all where, substantially high energy is involved. To enable accurate reconstruction of the comminuted fracture surgeon has to believe in his expertise and intuition to arrange the broken parts of bone and carry out the surgery. A puzzle solving technique will definitely help surgeons to practice reconstruction of the broken bone fragments prior to surgery in order to avoid errors in reconstruction. This paper presents a genetic algorithm based approach to obtain optimal sequence of bone reconstruction. In order to compare the performance of proposed approach, a well-known Iterative closest point algorithm is also implemented. The comparison is done based on computational time, mean deviation and number of iterations.**

Keywords—component; formatting; style; styling; insert (key words)

I. INTRODUCTION

Fracture reduction i.e. the task of repositioning the fragments of a broken bone into their original position is a common task in everyday orthopedics practice. For many fractures, the correct reposition is apparent and straight-forward to carry out in practice. However, an accurate reduction is difficult to achieve for some fractures because the preferred position of the fragments is difficult to infer from the existing medical images. The most widely studied fracture is the femoral shaft fracture. Figure 1 shows an example along with its modestly insidious treatment by intramedullary nailing [1]. In this procedure, a long nail is inserted into a bone via a small incision at the hip of knee. The fracture site is not directly visible, and surgeon has to rely on x-rays to align the fragments. This allows a nearly accurate repositioning in the image planes of the radiographs. The rotational alignment around the longitudinal axis of the bone poses a much greater challenge, as it cannot be observed in these radiographs. A clinical study done by Jaarsma and his group [2] has found a rotational malalingment of over 15 degrees in 28% patients.

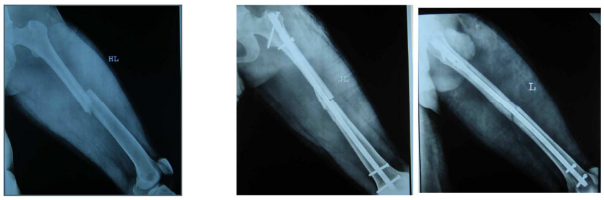

Fig. 1. A broken femur (left) treated with an intramedullary nail (right). Even in views from two perspectives, it is hard to judge the rotational alignment of the fragments [1]

II. PRIOR WORK

A few groups have worked on automatic bone fracture reduction. A computer assisted method, dealing with complex proximal humerus fracture was described by Bicknell et al [3]. They have reported an experimental study for the treatment of 4 part fracture via a hemiarthroplasty. They have compared CAS approach for placement of the hemoarthroplasty and the reduction of the tuberosity fragments with traditional surgecial technique. Another directly related work focuses on planning of fracture reduction of broken femur heads by Okada et al. [7]. They had proposed three different methods based on iterative closest point algorithm. The registration of fragments was done either (a) to the contralateral bone, (b) to fracture surfaces or (c) to fracture surfaces using the contralteral bone as a constraint. The authors concluded that first method suffered from the problem of local minima whereas other two overcome this problem. Meghari and Abolmaesumi [4] proposed an automatic method for global registration of multiple bone fragments. Their algorithm comprised of local and global registration steps. In the local step, each fragment was initially aligned to an anatomical plan of the bone based on statistical model. For each candidate pair a similarity transform was applied to match the fragment with the template. The point cloud of the corresponding parts were centered and oriented according to their eigen vectors and eigen values. After local registration, global registration was done using Kalman filtering. Their method of global alignment demonstrated high accuracy in a cadaver study. Willis et al [5] proposed a reconstruction technique for highly fragmented long bones by including, the segmented spongy bone. They have used multibody ICP algorithm with modified error metric. The method proposed byt them relied on manual

identification of likely corresponding fracture surfaces. Several aspects of this method were improved by Zhou et al. [6]. The manually recognized matching surfaces were structured into groups representing the same fragment pair in order to prevent oscillations in the pairwise registration. Additionally, geometrically stable subsampling was incorporated in the pairwise and multi-body alignment steps to improve the registration of featureless surfaces. The performance of the algorithm was demonstrated on one clinical tibila pilon case acquired from CT as well as on several artificially generated fractures based on bone replicas. Other variants of ICP used for automated data inspection [9], registration of 3D data [10] and fusing laser and vision data with genetic ICP algorithms [11] are the remarkable contributions to wards the performance enhanacement of the ICP algorithm.

III. PROPOPOSED APPROACH

The approach proposed in this paper is designed in a modular fashion as illustrated in Fig 2. The input data for the first module are CT scans of patient's fractured bone and mirrored contra lateral bone. The fragments of the broken bone are represented as surface meshes generated from the segmented CT scans. The main idea is to align main and functionally important fragments to an intact reference bone. This ensures anatomically correct repositioning of the main fragments, independent of the geometry of the fracture surfaces and possible small additional fragments. The contralateral bone is considered as a template and fragments of the bone are aligned to it using iterative closest point algorithm and genetic algorithm.

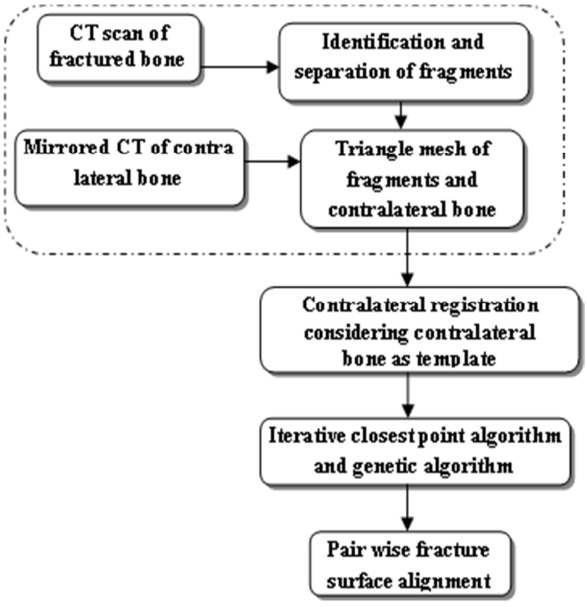

Figure 2 Proposed approach for fracture reduction

A. Iterative Closest point algorithm:

Iterative Closest Point (ICP) (Besl and McKay, 1992) [8] is an algorithm employed to minimize the difference between two clouds of points. ICP is often used to reconstruct 2D or 3D surfaces from different scans, to localize robots and achieve optimal path planning (especially when wheel odometry is unreliable due to slippery terrain), to co-register bone models, etc. The algorithm is conceptually simple and is commonly used in real-time. It iteratively revises the transformation (translation, rotation) needed to minimize the distance between the points of two raw scans. Inputs to the algorithm are points from two raw scans, initial estimation of the transformation and criteria for stopping the iteration. The output of the algorithm is the refined transformation.

The iterative closest point algorithm is developed to register two given sets of points or 3D shapes in a common coordinate system. The algorithm iteratively calculates the registration. In each iteration step, the algorithm selects the closest points as correspondences and calculates rotation and translation (R,t), for minimizing the following error equation

$$E(R,t) = \sum_{i=1}^{Nm} \sum_{j=1}^{Np} w_{ij} \left\| m_i - (Rp_j + t) \right\|^2 \qquad (1)$$

N_m and N_p are the number of points in model set M and dataset P, respectively. w_{ij} are the weights for a point match. They are assigned as $w_{ij}=1$, if m_i is the closest point to p_j and $w_{ij}=0$ otherwise. In the iterative closest point algorithm, transformations can be calculated by different methods. The version of ICP used in this work makes use of quaternions that can be used for two and three dimensions. Besl and McKay mathematically proved that ICP converges to a local minimum and hence to reach to global minimum, good initial feasible solution or initialization parameters are necessary. Local minima become an issue of major concern when points are overlapping partially or fully. The protocol followed by ICP is explained in the following figure 3. The point set of model and data to be checked are given. First estimation of the transformation is done. This estimation is considered as an initial value as well as maximally allowed error, the threshold ε and a boundary condition for non-convergence and maximum number of iterations, k_{max}. The figure 3(a) represents an initial state represented by the blue model point set and the red data point set. Figure 3(b) shows the search of closest point in the model to each point of data set. Then the new transformation matrices (fig. 3(c)) is calculated according to the minimized error of equation (1) and applied to the data set. Further, actual error is compared to ε. Iterations are continued until termination condition is achieved. The initialization process can be terminated as new improved state is obtained. Hence, in next iteration starts with the matching step as shown in fig. 3(d). These iterations are repeated until error is small enough as shown in fig. 3(e) or the maximum number of iteration k_{max} is reached. Even though algorithm converges at minimum error it is a best possible match rather than a perfect match.

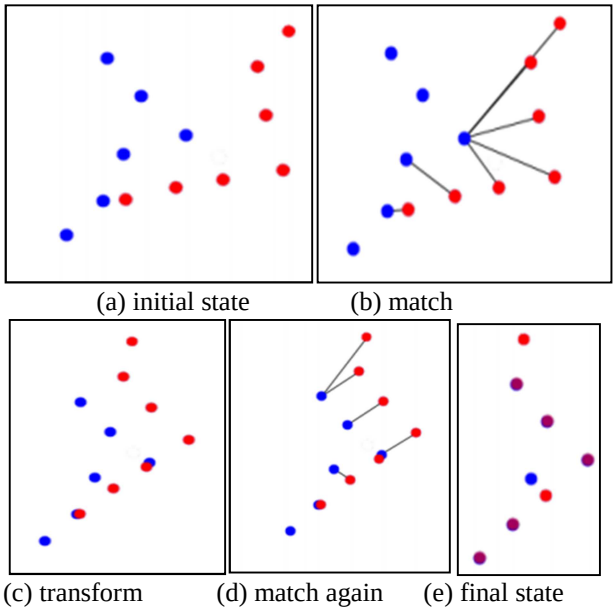

Figure 3 (a) Initial state with two point sets, the blue model and the red data set (b) Searching the closest points of each data point (c) State after the applied transformation (d) Next iteration, searching for the closest points again (e) Final matched state after the transformation

GENETIC ALGORITHM:

The iterative closest point algorithm is effective provided that a good initial feasible solution is available. If the initial solution is far away from the actual solution, then there are more chances of getting incorrect solution. In such one to one correspondence matching approaches, relations are determined by matching features taken from the images. However, since it is not possible to define a unique feature in 3D objects, correspondence matching depends on application under consideration. The correspondence matching can be done manually which might be very time consuming process. This in turn will make automatic surface registration impossible. This matching of two free-form surfaces can be framed as an optimization problem. formulation of such search problem may lead to a 6 dimensional optimization problem with many local extrema. This problem is proposed to be solved using simple genetic algorithm in this work. the mechanism used by genetic algorithm is based on natural selection and natural genetics. A possible solution is formulated as a chromosome in a string structure. Each element of this string structure represents one parameter in the solution. A collection of all possible solutions (chromosomes) forms a generation. It produces another generation through a search process. The search process of genetic algorithm follows the rule "survival for fittest". This rule is followed after a structured but randomized information exchange within the existing generation to yield a new generation. For the genetic algorithms to be successful, the methodology adopted to formulate the chromosome and fitness function is very vital. The genetic algorithms show good convergence provided that the continuity of fitness function is maintained. Moreover, the chromosome with the optimal fitness value must correspond to the target solution. The procedure adopted for formulations of the chromosomes and the fitness function for surface registration is described in the following section:

GENE AND CHROMOSOME

The geometric relation i.e. transformations between two surfaces can be defined by six parameters i.e. three parameters for position and three for orientation. These six parameters can be used to define as a chromosome. Because of this split in 6 parameters, 2 genes can be obtained i.e. translation gene which consists of transformation matrix along x,y and z direction and the rotation matrix which indicated rotation about x,y, and z axis. The structure of genes is as follows:

Translation Gene :
Tx: Translate on x axis ;
Ty: Translate on y axis ;
Tz: Translate on z axis

Rotation Gene :
α: Rotate about x axis;
β: Rotate about y axis;
θ: Rotate about z axis
S : scaling gene

Tx , Ty and Tz are the translation genes and, α, β and θ are the rotation genes. They form a chromosome [Tx, Ty, Tz, Rx, Ry, Rz] which represents the relation (3D transformation matrix) between two free-form surfaces, i.e the data points in two data sets are related by the mapping, T= Rx Ry Rz S , where

$$Rx = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\alpha & sin\alpha & 0 \\ 0 & -sin\alpha & cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$Ry = \begin{bmatrix} cos\beta & 0 & -sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ sin\beta & 0 & cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$Rz = \begin{bmatrix} cos\theta & sin\theta & 0 & 0 \\ -sin\theta & cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Tx & Ty & Tz & 1 \end{bmatrix};$$

FITNESS FUNCTION

A genetic algorithm uses a fitness function to determine the performance of each artificially created chromosome. Therefore the fitness function measures the registration quality i.e. the matching error caused by each chromosome. When two pairs match, the Euclidean distance between each correspondence pair tends to zero. Hence, for registration of two surfaces, genetic algorithm

searches for the minimum Euclidean distance between each correspondence pair. However, for true matching, determination of the final transformation is necessary. Hence, the "best possible" correspondence is used instead, to measure the fitness of a given chromosome, T. Given a set of points {Pi} in S_1(size N1) and {Qj} in S_2 (size N2), the point CPi in S_2 is defined as the "best possible" correspondence of Pi under the transformation T, such that the Euclidean distance, Ei between CPi and TPi is the minimum among all points in S_2. This is the best possible matching as any other correspondence will result into higher matching error. Since the two given free-form surfaces may not totally overlap each other, some points on surface S_1 may have no correspondence on surface S_2 even the identified transformation is correct. Therefore, if all Ei are considered, the fitness of the solution may not tend to zero as no correspondence can be found for some points. Therefore, the median of Ei as the fitness measurement is adopted instead of the average Euclidean distance Ei. So for a chromosome representing a transformation T, the corresponding fitness measurement is F(T) and is defined as: $F(T) = \text{Median}(Ei)$ for $1 \le i \le N_1$ where $Ei = |TPi - CPi|$ and $CPi = Qk$ such that $|Qk - T(Pi)| \le |Qj - T(Pi)|$ for all j where $1 \le j \le N_2$. Evaluation of fitness function described above requires a search on the closest point from a data set given an input data point.

REPRODUCTION

The reproduction state of genetic algorithm is about generation of a new set of possible solutions from the current set. For this purpose, Cross-over and Mutation are the generally used standard operators. In this formulation, real value coding of genes is done. Therefore, each gene in a chromosome is having a small value instead of changing from 0 to 1 or 1 to 0 for a binary gene during the mutation stage. The value of gene is generated randomly within the range [-MV, +MV]. While the maximum value (MV) has been kept constant. If the fitness value is large, the chromosome is far away from the optima point. Hence, a far jump is needed to get to a better chromosome and hence MV is kept at a larger value. Conversely, if only small movement is needed then MV is set to be a small value. Therefore, maximum allowed movement of the translation genes is set to FIT(Ti) / sqrt (3).

IV. CASE STUDY

In order to evaluate the performance of the proposed approach, one case problem of femur shaft fracture is considered. For this purpose, 3D solid model of femur shaft was taken (Fig. 4).

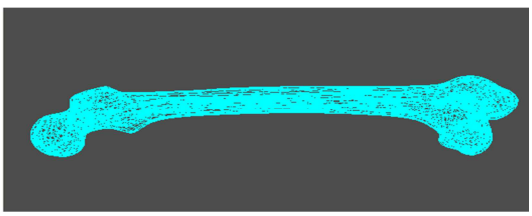

Figure 4 3D solid model of femur

A random fracture was introduced in the solid model. The 3D solid model of femur and fracture femur was converted to a .step file. The point cloud necessary for analysis was obtained from this file. The recognition of fragments of the bone was done manually. S_1 matrix consists of pose data of healthy bone and S_2 matrix consist of pose data of fragments. These both matrices were utilized as input to the iterative closest point algorithm and genetic algorithm. In order to reduce the computational burden, the point cloud of S_1 was of only 130 points where as for S_2 , it varies from 11 to 45 points based on size of the bone fragment. The plot of point cloud in 2d is presented in the Figure 5.

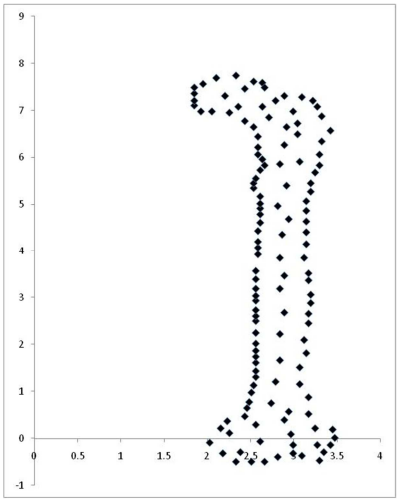

Figure 5 Plot of the 130 points obtained from the solid model for analysis (S_1)

The five different fragments were considered for the analysis. A solid model of individual fragment was obtained and meshed to get the point cloud. The number of cloud points considered for various fragments are as follows: Fragment 1 : 11 points, fragment 2: 11 points, fragment 3: 12 points, fragment 4: 12 points , fragment 5: 45 points. The points on the intact bone were used as target whereas points on fragments were considered source. The correspondence between these points cloud was obtained using iterative closest point algorithm and genetic algorithm.

V. RESULTS AND DISCUSSION

The above mentioned case problem was solved using Iterative closets point. This algorithm finds out the correspondence between two point clouds. ICP fit points in data (S_2) to the points in model (S_1). The fitting is done in such a way that the sum of square of errors with the closest model points and data points is minimized. The output of this algorithm is the rotation matrix and translation vector. If the original data points are multiplied by this rotation matrix and translation vector, then it gives the new data. This new data has correspondence with the model data (S_1). For this problem, maximum number of iterations were fixed to 1000, minimum number of iteration were set to 10. The fitting parameter was

considered to be 0.95 which means that fitting of data is done until 0.95 times of closest square errors are obtained. The error difference threshold for stopping iteration was considered to be 0.1. The meshing of solid model is done using coarse mesh and very few data points were used for analysis to reduce the computational burden. Hence, the threshold value was considered to be 0.1. The results obtained after implementation of this algorithm to the given problem using MATLAB are presented in the following Table 1.

Table 1 Performance of iterative closest point algorithm

Sr no	Fragment	Number of iteration	Mean deviation (New_data-old_data) New_data = R*old_data+T	Computational time (seconds)
1	Fragment 1	42	0.2577	22
2	Fragment 2	36	0.3461	23
3	Fragment 3	118	0.4792	68
4	Fragment 4	107	0.2148	48
5	Fragment 5	817	0.6138	130

SIMPLE GENETIC ALGORITHM:

The methodology of genetic algorithm presented in the previous section is implemented to obtain the correspondence between S1 and S_2 fragments. The pose data i.e. position and orientation of point is given as an input to the genetic algorithm. The value of one gene is computed by suitably substituting this pose data in T=[Rx Ry Rz S]. This is the value encoding of the gene. The value of crossover was set to 0.58. The results obtained after application of simple genetic algorithm are presented in the Table 2:

Table 2 performance of simple genetic algorithm

Sr no	Fragment	Number of generation	Mean deviation (New_data-old_data) New_data = R*old_data+T	Computational time (seconds)
1	Fragment 1	66	6.93	29.51
2	Fragment 2	49	1.81	39.92
3	Fragment 3	118	6.35	86.28
4	Fragment 4	130	6.56	68.78
5	Fragment 5	819	8.56	146.79

From the results presented in Table 1 and 2, it can be seen that the number generations/iterations required to bring correspondence between points on S_1 and S_2 by iterative closest point algorithm are less as compared to genetic algorithm. The main reason for this observation can be related to random search technique adopted by simple genetic algorithm. The number of generations required can be reduced if the guess solution is predicted very near to the desired solution. Moreover, it can be seen that because of more number of generations, computational time required for the genetic algorithm is also more. The analysis is performed using intel core-2 duo processor with 1.60 GHz) speed. The obvious reason is the number of iterations and number of data points handled.

The mean deviation is nothing but the difference between the actual data and the data obtained using algorithms. The results obtained using genetic algorithm are not impressive as compared to iterative closest point algorithm. A higher value of mismatch is presented by the standard deviation values for genetic algorithm whereas ICP gives better results.

The major difference between two methodologies lies in the concept. The simple genetic algorithm makes use of position as well as orientation of the point whereas ICP utilizes only position information of any point. Because of this, the amount of information handled by ICP is less as compared to genetic algorithm. Moreover, number of points in the point cloud considered for study is less as compared with the datapoints obtained after meshing. The accuracy of fragment correspondence can be increased if more number of data points are considered. The performance of GA can be further increased if hybridization of ICP is done with GA. This will result in utilization of ICP for prediction of initial guess solution and improvement of solution will be done by the genetic algorithm.

From above analysis, a rough estimate of sequence of fracture reduction and quality of output can be made. However, in order to provide a complete solution of fracture reduction can be provided only if 1. Complete CT scan data is utilized 2. Presentation of algorithm output is done in an interactive fashion and 3. User friendly interface is made.

VI. CONCLUSIONS

This paper presents a concept of implementation of simple genetic algorithm to get the proper sequence and position of the fracture fragments in correspondence with the original bone. The performance of the proposed genetic algorithm is compared with iterative closest point algorithm. From above discussion, it can be concluded that the genetic algorithm can handle pose data of any point comfortably in contrast with ICP. This aspect is most important to present the outcome of algorithm in an interactive fashion. From computational point of view, genetic algorithm is costly. However, implementation of any methodology can improve the performance of genetic algorithm.

REFERENCES

[1] T. Albrecht and T. Vetter, "Automatic fracture reduction, MeshMed", A book by Springer Verlag, Berlin, pp 22-29, 2012.

[2] R. Jaarsma and D. Pakvis, "A rotational malalignment after intermedullary nailing of femoral fractures", Journal of

orthopedic trauma, vol 18, no 7, pp 403-410, 2004.
[3] R.T. Bicknell, "Early experience with computer assisted shoulder hemiarthroplasty for fractures of the proximal hmerus; development of novel technique and in vitro comparison with traditional methods", Journal of Shoulder and Elbow Surgery", vol 16, no 3, pp 117-125, 2007.
[4] M.H. Meghari, and P. Abolmaesumi, "Global registration of multiple bone fragments using statistical atlas models, feasibility experiments", Engineering in Medicine and Biology Society, pp 5374-5377, 2007.
[5] A. Willis, "3D reconstruction of highly fragmented bone fractures", SPIE proceedings, vol 6512, 2007.
[6] B. Zhou, "Virtual 3D bone fracture reconstruction via inter-fragmentary surface alignment, IEEE workshop on 3D digital imaging and modeling", IEEE Computer Society, 2009.
[7] T. Okada, "Computer assisted pre-operative planning for reduction of proximal femur using 3D CT data", Biomedical Engineering, IEEE Transaction, vol 56, no 3, pp749-759, 2009.
[8] P. Besl and H. McKay, "A method for registration of 3D shapes", IEEE Transactions on pattern analysis and machine intelligence, vol 14, no 2, pp 239-256, 1992.
[9] Q. Shi and N. Xi, " Automated data processing for a rapid 3D Surface Inspection System, Robotics and Automation (ICRA)", IEEE International Conference, pp.3939-3944, May 2008.
[10] T. Ridene and F. Goulette, "Registration of xed-and-mobile-based terrestrial Laser data sets with DSM", Computational Intelligence in Robotics and Automation (CIRA), IEEE International Symposium on, pp.375-380, December 2009
[11] Q. Muhlbauer, "Fusing laser and vision data with a genetic ICP algorithm", 10th international conference on IEEE ICARCV, Hanoi, pp. 1844-1849, 2008.