Investigating Evolutionary Optimisation of Constrained Functions to Capture Shape Descriptions from Range Data

C.Robertson[1], **R.B.Fisher**[1], **D. Corne**[2], **N. Werghi**[1], and **A. Ashbrook**[1]

[1] *Research Vision Group, Department of Artificial Intelligence,*
University of Edinburgh, 5 Forrest Hill, Edinburgh, EH1 2QL.
[2] *Parallel, Emergent and Distributed Architectures Laboratory, Department of*
Computer Science, University of Reading, Reading, Berkshire, RG6 6AY.

Keywords: Genetic algorithms, shape fitting, constrained fitting

Abstract

*This paper examines the application of an evolutionary algorithm (*GENOCOP III*) to the problem of fitting surfaces and lines to both 2D synthetic and real 3D range data. The fitting is performed with both non-linear (domain) constraints and with non-linear (geometric and relational) constraints. Example fittings are given as well as an explanation of experimantal caveats that should be addressed when using this kind of approach. We have discovered, for example, that the time to generate starting reference points on the solution manifold is a significant part of the experiment time.*

1. Introduction

Shape analysis of objects from range data (captured three dimensional coordinates of surface points) is a key problem in computer vision with several important applications in manufacturing, such as assembly, quality control and reverse-engineering. The problem is generally formulated as a nonlinear programming problem (NLP), which tries to optimally fit the data to candidate shape descriptions. The NLP optimises a function subject to several constraining equations and inequalities. Especially with nonlinear constraints, it is notoriously difficult to optimise and there is no known method to guarantee a satisfactory solution. Only if the optimisation function and the constraints are satisfied are the optimal point (or points) likely to be found. Traditional techniques, such as gradient descent, are unsatisfactory for the solution of NLPs, due to the local nature of their search methods and the reliance on smooth derivatives in the search-space. The formulation of constraints and the application of constraint-based correction and optimisation of surface fitting has been achieved previously [1] with notable success using the optim1 and optim2 constraint application strategies. There are, however, some associated problems with this approach, such as the complex formulation of the objective function and reliance on the convexity of the solution space. The first problem is addressed here by using only the least-squares metric as an objective and using the constraints to define the manifold of allowable solutions. In this work we examine the applicability of evolutionary strategies to the problem of fitting lines and surfaces to both synthetic and acquired object range data. This is a traditional NLP since a priori constraints exist between the fitted surfaces and lines. The GENOCOP III algorithm [2, Ch.7] was used in this paper. It is an evolutionary algorithm system which is specialised to handle constrained function optimisation. GENOCOP III uses real-valued genes, and included methods to deal with linear, non-linear, domain and inequality constraints. We have examined the use of GENOCOP III, with a specialised fitness function (described later), applied to the problems of fitting both 2- and 3- dimensional surface equation constructs to range data while simultaneously applying several necessary geometric and domain constraints. The constraints cover the two typical types : domain, the restriction on the parameter size; relational, relationships between surfaces that are known a priori.

2. Chromosome Representation and Fitness Functions

Fig. 2.1 is a typical example of the kind of data that is derived from a laser range-finder. The data is from a typical engineered part comprising a number of planes and degenerate quadric surfaces arranged in a fairly well defined way. There are, for example, several places where angles of 90^o have been used as relationships. There is also a cylinder at one end whose axis lies orthogonal to the normal of some planes and parallel to the normals of others. It is these kinds of relationships we are seeking to preserve.

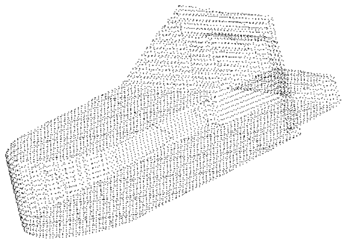


Fig. 2.1. Example 3d range data points

We use a real-valued chromsome representation in which each gene corresponds to a coefficient in a shape equation. In the cases we have investigated typical chromosomes are shown in Fig 2.2(a)-(c).

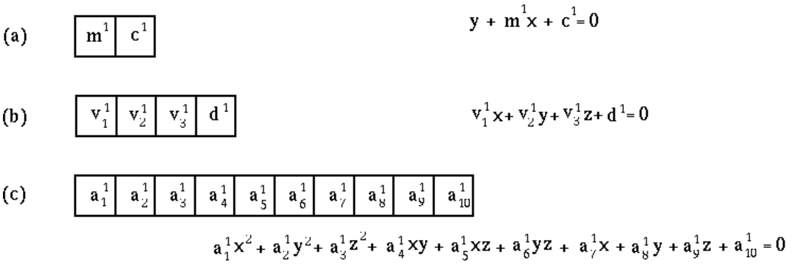
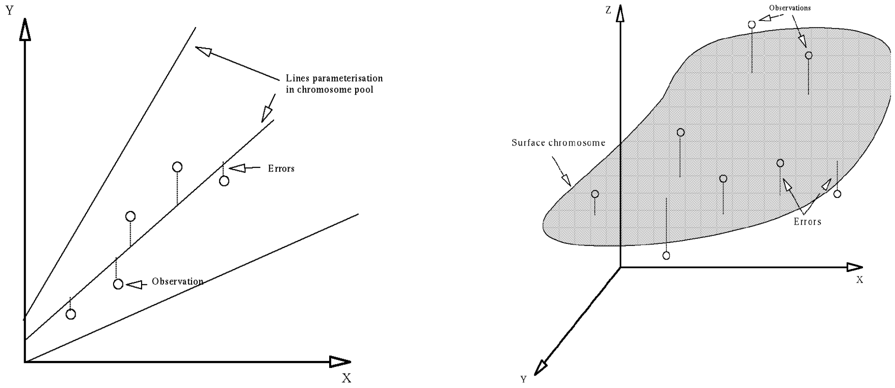

(a)
m1 c1
y + m1x + c1 = 0
(b)
v11 v21 v31 d1
v11x + v21y + v31z + d1 = 0
(c)
a11 a21 a31 a41 a51 a61 a71 a81 a91 a101
a11x2 + a21y2 + a31z2 + a41xy + a51xz + a61yz + a71x + a81y + a91z + a101 = 0

Fig. 2.2. Chromosomes representing (a) a line (b) a plane (c) a general quadric

The surfaces were fitted using a standard least-squares metric over a large number of surface points. The value at each point gives an indication of the probability that the point belongs to the parameterized surface; when summed this gives a good overall estimation of how well the parameterized surface matches the data.



(a) Lines *(b)* Surfaces

Fig. 2.3. Line and surface parameterization and error measures

For lines, the best line is defined as the least squares fit through the observation data:

$$E = min \sum_{i=1}^{i=N} (y_i - mx_i - c)^2 \tag{2.1}$$

In this case the parameterized line is (m,c) and the observation data is (xi, yi). In the case of two separate lines, where some constraint exists, this error is expressed as the sum of the two errors. In the case of planes, the least-squares error is given by :

$$E = min \sum_{i=1}^{i=N} (\mathbf{v}.\mathbf{x}_i + d)^2 = minF(\mathbf{v}, d) \tag{2.2}$$

where $\mathbf{v}$ is the plane parameterization and x_i is the observation range data. Notice that F is homogeneous with respect to the parameters of $\mathbf{v}$ and d. This means that the solution must be constrained in order to avoid the trivial null solution. The most natural constraint is $\mid v \mid = 1$, the unit normal, which also has the advantage of being invariant with respect to translations and rotations. For quadrics the geometric distance from a point the surface is non-trivial to compute so the most straightforward distance measure to use as an error measure is the algebraic distance:

$$E = min \sum_{i=1}^{i=N} (\mathbf{x}_i^t A \mathbf{x}_i + \mathbf{x}_i.\mathbf{v} + d)^2 = F(A, \mathbf{v}, d) \tag{2.3}$$

where the quadric parameters are given as a symmetric matrix A, the vector $\mathbf{v}$ and the constant where :

$$A = \begin{bmatrix} a_1 & a_4/2 & a_5/2 \\ a_4/2 & a_2 & a_6/2 \\ a_5/2 & a_6/2 & a_3 \end{bmatrix}$$
$$\mathbf{v} = [a_7, a_8, a_9]$$
$$d = a_{10}$$

Once again, it can be seen in equation 2.3 that the function F is homogeneous with respect to the parameters $a_1..a_{10}$. This means that the problem of fitting must be constrained to avoid the trivial null solution. The constraint we have employed is the same as suggested in [3], where the first six parameters of the quadric essentially the shape parameters) are constrained to be unity: $Tr(AA^t) = 1$. Again, this is invariant with respect to a rigid transformation, rotation or translation.

3. Geometric Constraints

In the real world all optimization problems are constrained. We need to employ constraints in our fitting for the following reasons :

- *Avoiding null solutions.* To avoid convergence of parameter values to zero, virtually every surface parameterization has a unit normal constraint or equivalent.
- *Improved fitting.* They improve the overall fit of the data since boundary constraints can be used to ensure that solutions do not fall outside a required range. This means that virtually every parameterization requires a set of inequality constraints.
- *Inclusion of a priori information.* For example, if two planes are known to be orthogonal then the chromosome represents a complex form as shown in Fig. 3.1. This means that although the two lines of best fit may be at some angle F to each other, the actual value may be known and used in order to fix the value at the correct one. Usual examples are $\phi = 90^o$ or

$\phi = 0^o$. Such errors often result from mis-registration of multiple views of data, which is also known to be a difficult problem. In the case of reverse-engineering, these kinds of constraints are very often known, whereas the parameterization of the actual system components are not.

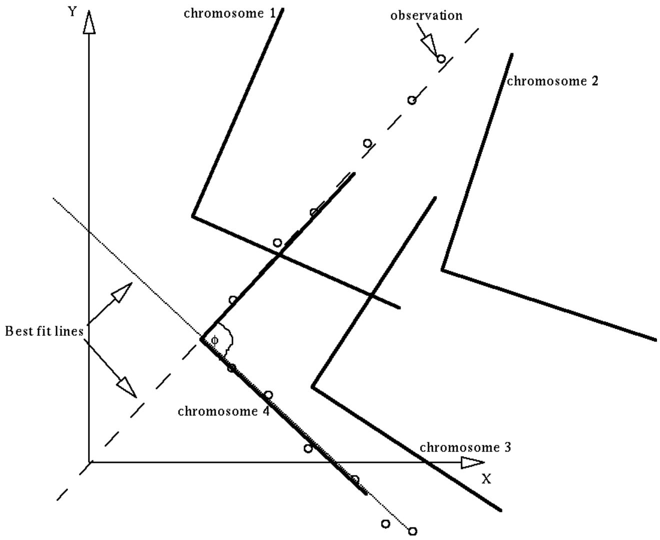


Fig. 3.1. Candidate chromosomes for an orthogonally-constrained line-fitting case

3.1 The GA Constraints Strategy

Overall flow of control in a GA normally consists of the following operations:

1. Random generation of initial population
2. Evaluation of fitness of chromosomes in the population
3. Application of genetic operators
4. Repeat 2-4 as required

The question of how constraints are implemented as a method of population shaping, (keeping solutions on the constraint manifold), is slightly more complex. If non-conforming chromosomes are simply culled, the populations can become severely reduced at each cycle, thus scuppering the diversity needed for effective search. A penalty function can be introduced into the fitness function which inhibits the growth of individuals straying from the constraints or dependent upon the number of constraints violated. This is the most common approach, but is difficult to implement effectively since the penalties for all constraints must be considered independently. The use of repair operators, to re-map infeasible chromosomes back to the feasible space has also become popular recently. It is sometimes (but not always) easy to repair chromosomes and those repaired may be used for evaluation or may be placed (with some probability) back into the population.

GENOCOP III uses a repair and replace heuristic as well as a specialized set of operators. In [2, Ch. 7.5], Michalewicz suggests an empirical figure of 15% replacement seems optimal for repaired individuals. GENOCOP III also uses operators that ensure that any mutation operation or crossover operation always produces a child that is from the constrained solution space. Example operations are:

- *Uniform mutation.* Where the value of a gene may mutate, but only into the range specified by its domain constraint.
- *Boundary mutation.* Where the mutation pushes the gene toward its domain boundary.
- *Arithmetical crossover.* This is the linear combination of two vectors such that closure is guaranteed.

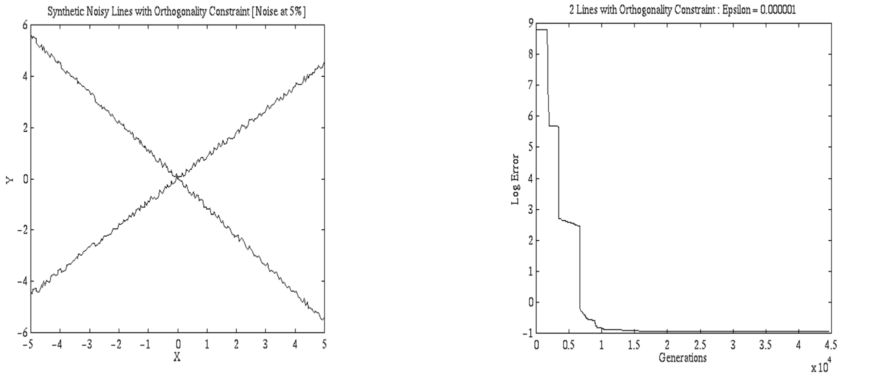
There are several other such operators. As well as these, GENOCOP III maintains two sets of populations, a search population and a set of reference points. The feasibility of the search population is maintained by using appropriate operators. A set of search points is also generated and repaired as necessary to provide new potential reference points. If the search point generated is better than one of the currently held reference points then it is replaced.

4. Experiments

This section details some of our experiences performing fittings for both 2D and 3D data with constraints. Note that the 3D points were all scaled into a unit sphere.

4.1 2D- Line Fitting with Constraints

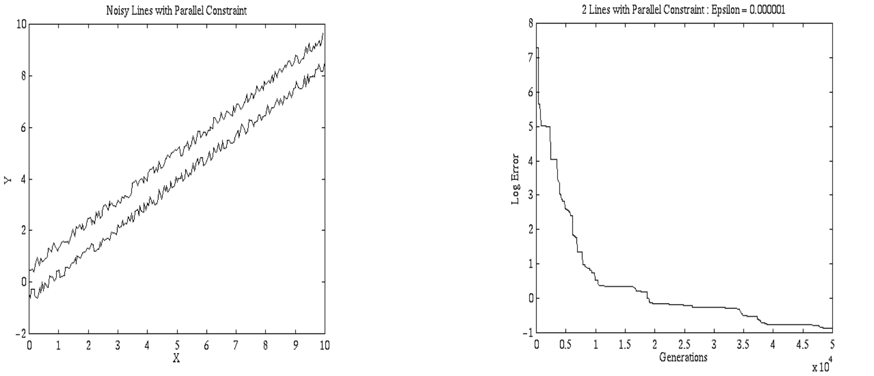
In the first instance, 2-dimensional linear features with orthogonality and parallel-ness were fitted to noisy synthetic data. With conventional techniques this is not a particularly hard problem when low numbers of lines are used. In our case we examined sets of two lines at various constrained angles to each other. Gaussian noise was applied to the data at varying values. In each case there were 400 (x,y) data points in two sets of 200. The initial population of 100 chromosomes can be seen to converge to a useful fit after about 35000 evaluations, taking around 12 seconds on an average load 200MHz Sun Ultrasparc 1. It should also be noted that convergence in the case of parallel lines was not made more difficult by the fact that local solutions where the two lines are the same were reached. Whilst constraints could have been applied



(a) Noisy constrained line *(b)* Orthogonally constrained lines

Fig. 4.1. Orthogonal Line Fitting

to avoid these solutions, it was not found necessary. In each case the degree to which the chromosome could stray from the constrained solution manifold was defined to be e = 0.000001 units. Example data and graphs are shown in figures 4.1 and 4.2.



(a) Synthetic lines with noise *(b)* Constrained lines

Fig. 4.2. Parallel Line Fitting

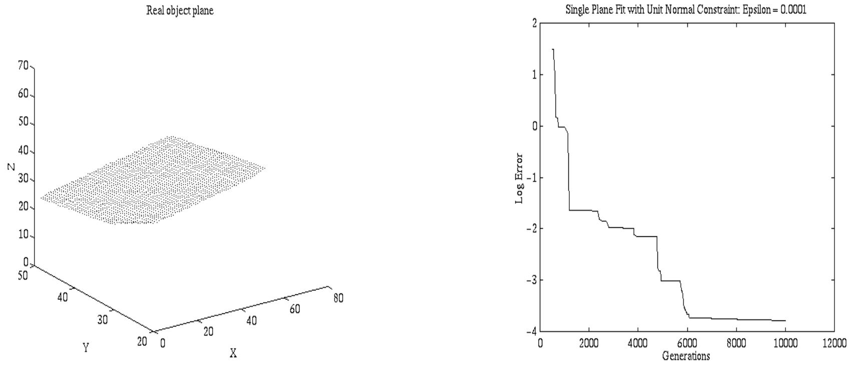
4.2 3D- Plane and Quadric Fitting with Constraints

In the 3-dimensional case the following sections describe the fittings that have been performed.

4.2.1 Plane with unit normal constraint. A single plane with a unit normal constraint, chromosome with 4 genes representing the plane parameters was fitted. This plane data is from a real object and has a error of around 0.3mm. In this case the following chromosome was used: v = (v1, .., v4), where (v1, v2, v3) represented the surface normal. The constraints were as follows:

1. $(v_1^2 + v_2^2 + v_3^2) = 1$
2. $\mid v_1 \mid, ..., \mid v_4 \mid < 1$

The original plane is shown in Fig. 4.3(a) and the surface fitting error is shown in Fig, 4.3(b). Experience has shown that in the case of plane fitting often the first parameter to stabalize is the constant, followed by the rotation of the normal. This can be seen in the error graph in Fig. 4.3(b) where the initial steep descent is the fixing of this parameter.



*(a)*Plane from real object

*(b)*Errors at each evaluation step

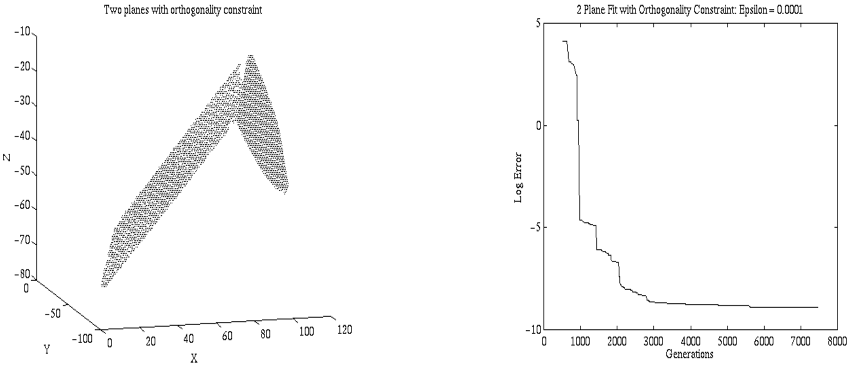
Fig. 4.3. Plane fitting

4.2.2 Two Planes with Angle Constraint. Two planes with an orthogonality constraint and two unit normal constraints, chromosome with 8 genes representing the 2 plane parameters, was fitted. In this case, the chromosome was $v = (v_1, .., v_8)$, where (v_1,v_2,v_3) and (v_5,v_6,v_7) were the surface normals. The constraints were as follows:

1. $(v_1^2 + v_2^2 + v_3^2) = 1$
2. $(v_5^2 + v_6^2 + v_7^2) = 1$
3. $v_1 * v_5 + v_2 * v_6 + v_3 * v_7 = 0$ - two plane orthogonality constraint
4. $\mid v_1 \mid, ..., \mid v_8 \mid < 1$

It can be seen from the graph in Fig. 4.4(b) that the errors in the fitting decrease to a very good level after only 2000 evaluations.

4.2.3 Degenerate General Quadric with Unit Constraint. A general quadric fit of a sphere with a unit constraint on the first six parameters, chromosome of length 10 genes representing the quadric parameters, was fitted. In this case there was no explicit use of the fact that the data represented was from a sphere.



(a) Planes from real object *(b)* Errors

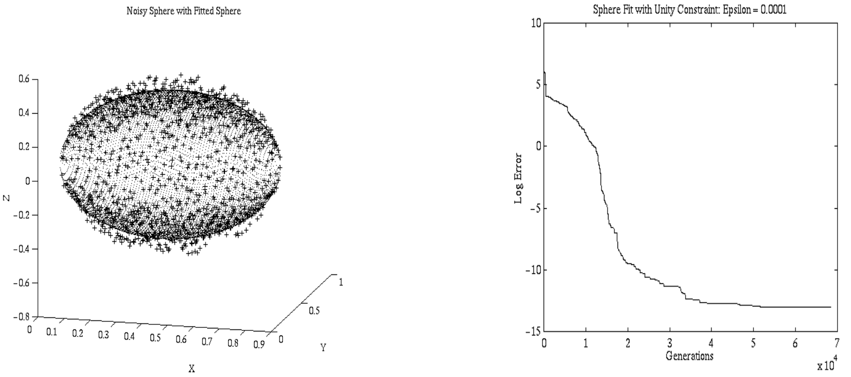
Fig. 4.4. Two constrained planes fitting with orthogonality constraint

In this case, the chromosome was $a = (a_1, .., a_{10})$ where each gene was one of the quadric parameters. The constraints used were as follows :

1. $(a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2 + a_6^2) = 1$ -unity constraint to avoid null solution

2. $\mid a_1 \mid, ..., \mid a_{10} \mid < 1$

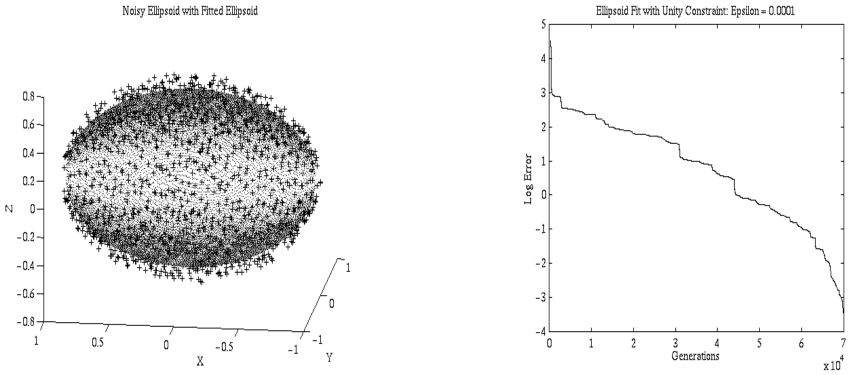
After the constant terms found a good approximation to their best values each of the shape and pose genes $(a_1, ..., a_6)$ in the fittest solution chromosome converged gradually to their optimal value.



(a) sphere (+) and fitted sphere (.) *(b)* Fitting errors

Fig. 4.5. Noisy synthetic sphere fitting

4.2.4 General Quadric (Ellipsoid) with Unit Constraint. A general quadric fit of an ellipsoid with a unit constraint on the first six parameters, chromosome of length 10 genes representing the quadric parameters. The methodology was as in the last section. The data is shown in Fig. 4.6(a) and the resultant graph of fitting errors for the winning chromosome at each evaluation is shown in Fig. 4.6(b).

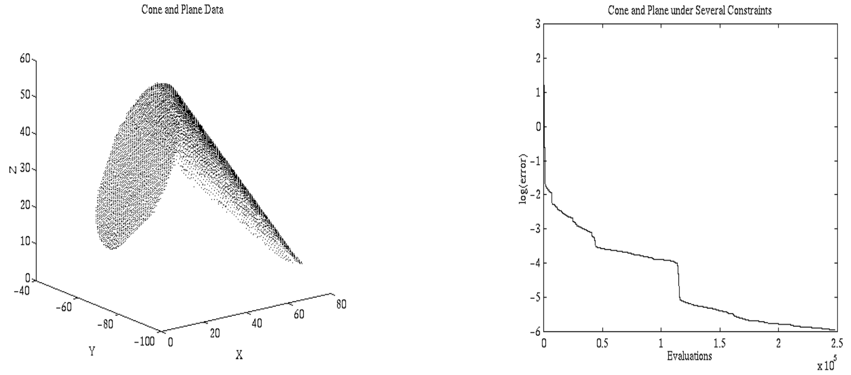


(a) ellipsoid (+) fitted sphere (.)

(b) Fitting errors

Fig. 4.6. Noisy synthetic ellipsoid fitting

4.2.5 Compound - Degenerate Quadric and Plane with Geometric Constraints. In this experiment we fitted a cone with a planar bottom, where the cone is constrained to be both right and circular with axis along the normal of the plane. A chromosome of 14 genes represents the quadric parameters of the cone and the parameters of the plane, $G = (v_1, .., v_4, a_1, ..., a_{10})$, where (v_1, v_2, v_3) is the normal of the plane, v_4 is a constant and $a_1, ..., a_{10}$ are the quadric parameters.



(a) Real object data

(b) Fitting errors

Fig. 4.7. Cone and plane fitting

The constraints used in fitting the two surfaces in this object were::

1. $(a_1^2+a_2^2+a_3^2+a_4^2+a_5^2+a_6^2) = 1$ -unity constraints to avoid null solution
2. $(v_1^2 + v_2^2 + v_3^2) = 1$
3. $a_1 - a_2 = v_1^2 - v_2^2$ - circularity constraints
4. $a_1 - a_3 = v_1^2 - v_3^2$
5. $a_2 - a_3 = v_2^2 - v_3^2$
6. $a_4 = v_1.v_2$
7. $a_5 = v_1.v_3$
8. $a_6 = v_2.v_3$
9. $|v_1|, ..., |v_4|, |a_1|, ..., |a_{10}| < 1$

4.3 Experiment Summary and Caveats

The following table summarises the experiements performed, where 'DC' stands for 'Domain Constraints' and 'GC' stands for 'Geometric Constraints'.

Type	No. points	e value	DC	GC	Evaluations
Lines	400	0.000001	4	1	30,000
Plane	2217	0.0001	4	1	10,000
Two planes	4913	0.0001	8	3	5,000
Sphere	1941	0.0001	10	1	70,000
Ellipsoid	1776	0.0001	10	1	70,000
Cone + plane	8199	0.0001	14	8	250,000

One typical problem is that given a large chromosome with many parameters it is often particularly hard to find an initial start condition unless the original least-squares fitting for each of the surfaces is available. Even if so, a start-point may sometimes not be readily calculated. Several fixes for this were tried, for example using vector cross-products to ensure orthogonality, but in the future a more rigorous approach will be formulated. Another problem was that given an arbitrary quadric surface from a range image, the data origin may be some distance from the origin. This means that the traversal distance in the search-space of the last 4 parameters of the quadric is high relative to the first six (which are unit normalized). Often this makes a satisfactory solution unlikely. In order to achieve a fitting, all of the sampled range data were normalized to a unit sphere, centred at the origin, and a scale factor extracted. This proved a good remedy since the traversal distances for all parameters was at least the same order of magnitude.

5. Conclusions

First and foremost, this paper represents a proof of concept - an evolutionary algorithm can be used to fit surfaces where a priori domain and geometric constraints are important. All of the fittings in the last section satisfied both the constraints and the least-squares fit to a high degree. This initial success has indicated to us that evolutionary strategies are feasible in this domain, but the time complexity to convergence (each fitness evaluation involved calculations for each of several thousand coordinate pairs or triplets) has suggested possible poor scaling to larger and more complex surfaces. To deal with these we did further experiments which involved evaluating candidates in early stages against only a small proportion of the available data. This succeeded in significantly speeding up the process without unduly misleading the future direction of search. One other approach we are investigating involves the off-line computation of datasets and parameterisations which has been shown [1] to significantly speed up least-squares fitting. Work so far thus indicates the viability of evolutionary approaches to fitting range data from complex real-world components with engineered surfaces. Future work in this area is as follows:

- *Segmentation.* We are assessing whether a 3-dimensional segmentation of the data (that is the decision of which points belong to which surfaces) may also possible using a priori constraints on the range of possible shapes and their relationships.
- *Failure points.* We are assessing at what point the system breaks down under the constraint loading and surface complexity.
- *Feature identification.* We are examining the explicit coding of constraints for surface features, such as a slots or a drilled hole. This is a particularly difficult problem in 3D segmentation because often sufficient surface data does not exist to segment each of the individual elements of, say, a drilled hole. If constraints are taken into consideration then it may be possible to code the individual feature elements into a single chromosome and search for it in a scene.

Acknowledgments

This work was funded by UK EPSRC grant GR/H86986. GENOCOP III is available from Michalewicz's ftp site at ftp://ftp.uncc.edu/coe/evol . Updates are also available from http://www.coe.uncc.edu/ gnazhiya/gchome.html.

References

1. Werghi, N., Fisher, R.B., Ashbrook, A., Robertson, C. (1997): Improving model shape acquisition by incorporating geometric constraints. Proc. British Machine Vision Conference BMVC97, Essex, 520–529
2. Michalewicz, Z. (1996): Genetic Algorithms + Data Structures = Evolution Programs. Springer, 3rd edition
3. Faugeras, O., Hebert, M. (1986): The Representation, Recognition and Positioning of 3-D Shapes from Range Data, in Techniques for 3-D Machine Perception, ed. A. Rosenfeld, North-Holland, Amsterdam