Using Multiple Genetic Algorithms to Generate Radar Point-Scatterer Models

Evan J Hughes and Maurice Leyland

Abstract— **This paper covers the use of three different genetic algorithms applied sequentially to radar cross section data to generate point scatterer models. The aim is to provide automatic conversion of measured 2D/3D data of low, medium, or high resolution into scatterer models. The resulting models are intended for use in a missile–target engagement simulator. The first genetic algorithm algorithm uses multiple species to locate the scattering centres. The second and third algorithms are for model fine tuning and optimisation respectively. Both of these algorithms use non-dominated ranking to generate Pareto-optimal sets of results. The ability to choose results from the Pareto sets allows the designer some flexibility in the creation of the model. A method for constructing compound models to produce full 4π steradian coverage is detailed. Example results from the model generation process are presented.**

Keywords— **Multiple Genetic Algorithms, Pareto Optimal, Multiple Species, Radar Cross Section, Point Scatterer Models.**

I. INTRODUCTION

MODERN missile design techniques based upon genetic algorithms (GAs) and neural networks often require fast and accurate missile–target engagement simulations. Many simulation systems use random numbers to mimic the effects of a fluctuating target radar cross section (RCS) in an attempt to minimise simulation times. The RCS models described in this paper are created from real RCS data but still retain the rapid execution times required for simulation. Real high-resolution target RCS data consume massive amounts of disk storage space. The storage requirement may be overcome by processing the data off-line to generate a model that can be used to recreate quickly a good approximation of the original RCS.

The model may be produced from the source data by identifying and storing the locations of the main reflective surfaces of the target. These collections of scattering centres can be replaced by models consisting of ideal isotropic point scatterers placed in the same locations. These models approximate the target sufficiently for our purposes. Point scatterer models allow the effective resolution of the data to be increased by interpolating between measured sample points. The interpolation is nonlinear and is related to the arrangement of the scatterers. The interpolated data therefore appear as a realistic RCS pattern.

Traditionally, the conversion process is very computationally intensive. The GA system described in this paper automates the conversion of RCS data into realistic models and reduces the processing overhead involved in the conversion process. The GA approach makes it practical to

generate models using only a desktop PC. The paper finally describes the construction of compound models that give full 4π steradian coverage and still retain the rapid execution properties of simple point-scatterer models.

II. CONVERSION PROCESS

Figure 1 shows a block diagram of the conversion process. The RCS data need to be measured over a small range of azimuth and elevation angles and for a spread of frequencies. These data can then be used to form a three-dimensional *inverse synthetic aperture radar* (ISAR) image [1, Page 726]. This image is analogous to an optical hologram and allows the rough spatial locations of the major scattering centres to be identified [2].
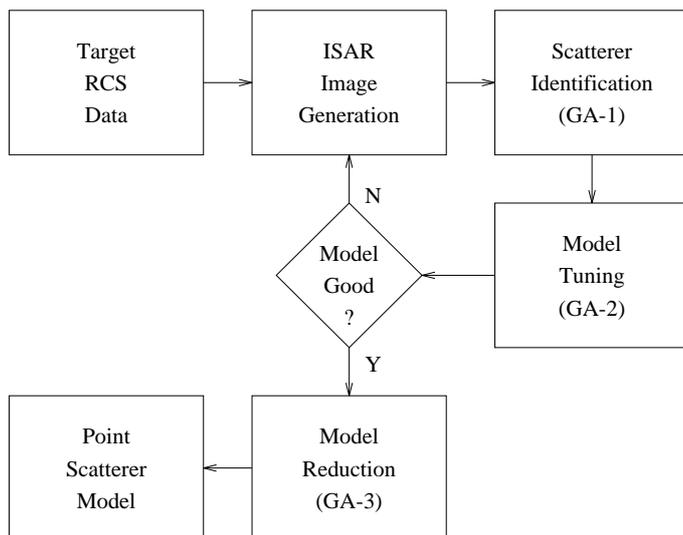

Fig. 1. Block diagram of RCS data conversion process

The positions of the scatterers are located using the first GA, which has a population split into multiple species and is capable of identifying multiple scatterers in each run. A model is generated with scatterers located at the rough positions identified from the image and the second GA is applied. This algorithm is designed to fine tune the location of the scatterers to improve the accuracy of the model.

The RCS of the model is checked against the required target pattern. If the model does not fit, an ISAR image of the model is generated. This image is then subtracted from the original image. The subtraction reveals the scattering centres that have not yet been included in the model. The identification/fine tune cycle is repeated until the RCS of the model matches the required data satisfactorily. At this point, the model may have well in excess of 100 scatterers.

The authors are with the Department of Aerospace, Power and Sensors, Royal Military College of Science, Cranfield University, Shrivenham, England. E-mail: ejhughes@iee.org .

Once satisfactory model elements have been generated when compared to the ISAR image and RCS data, the model must be reduced to a convenient size and fidelity. The third GA performs a combinatorial search of different numbers of scatterers and configurations in an attempt to reduce the number of scatterers in the model whilst minimising the induced error. A *Pareto-optimal* [3, Pages 197–201] set of evaluated solutions is then generated. The Pareto set allows the designer to trade between final model size and the accuracy of the RCS to the original data. Final models suitable for our simulations often contain around 100 scatterers.

The RCS of a point scatterer model at wavelength λ may be defined as:

$$\rho_T = \left| \sum_{k=1}^{n} \sqrt{\rho_k}\, e^{j\left(\frac{4\pi d_k}{\lambda}\right)} \right|^2 \qquad (1)$$

The total RCS of the target, ρ_T, is defined as the square modulus of the coherent sum of the echos from the n scatterers. Each scatterer has its own RCS, ρ_k, and is at a distance d_k from the observation point [1, Page 23]. The sum of the echos is a complex quantity with units of volts. The RCS is a scalar with units of square metres.

The effective scaling of ISAR images in cross-range (up–down, left–right) and slant range (front–back) is determined by the sample step size and number of samples in azimuth, elevation, and frequency. Eq. 2 and 3 show how to calculate cross range and slant range resolution respectively.

$$
\begin{aligned}
r_c &= \frac{c}{2 n_c \delta \theta \overline{f}} \qquad (2) \\
w_c &= n_c r_c
\end{aligned}
$$

$$
\begin{aligned}
r_s &= \frac{c}{2 n_s \delta f} \qquad (3) \\
w_s &= n_s r_s
\end{aligned}
$$

where n_c is the number of steps in cross range, n_s is the number of steps in slant range, $\delta\theta$ is the angular step size in radians, δf is the frequency step in Hertz, c is the speed of propagation in metres/sec., and $\overline{f}$ is the mean frequency in Hertz. The cross range and slant range resolutions are denoted by r_c and r_s while the total range extent are w_c and w_s respectively. All range measurements are in metres. Care must be exercised in the choice of angular window ($n_c \delta \theta$) that the measurements are taken over. A window greater than $10°$ will begin to cause a blurring at the edges of the image. Techniques can be used to focus the image, thereby reducing the burden on the fitting process. Due to the wider angle of interest, more scatterers are eventually required to fit the RCS to the model accurately. Image generation involves the application of Fourier transforms to the RCS data and therefore suffers from the spectral spreading problems inherent in this process. The GAs do not require the application of window shaping functions to reduce spreading effects. The inherent square window leads to sharp peaks with long tails extending into the image. This actually aids the search abilities of the first GA (section III-C).

III. Scatterer Identification

A. Introduction

An iterative method may be used to identify the scatterer locations [4]. This method operates by first finding the size and coordinates of the brightest spot in the image and places a scatterer in the corresponding position in the model. An approximation of the ISAR image of the scatterer is then subtracted from the original image to remove the corresponding bright spot. The process is repeated until all the major bright spots have been removed. Figure 2 illustrates the algorithm. Other methods have been developed that rely on drawing contour maps of the image and locating the scatterers within the bounded regions [5].
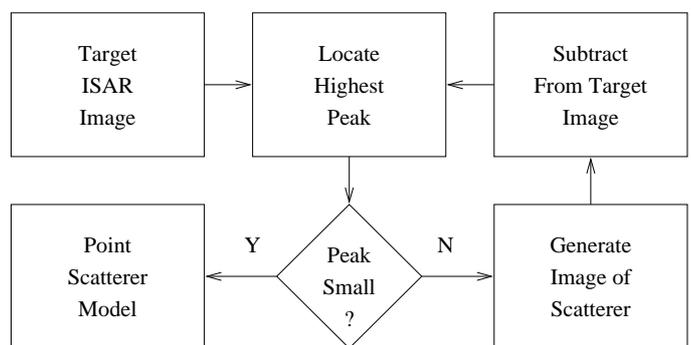

Fig. 2. Iterative RCS data conversion process

Those methods work well but have one major drawback: they require high-resolution data in order to locate the centre of each scatterer accurately. High-resolution data is easily obtainable if the target is computer generated from a CAD model, but it is not practical to analyse a real target in such fine detail. A typical three-dimensional image of 640 pixels on each axis will require $(640)^3$ elements and therefore two gigabytes (Gbytes) of storage space. Finding the location of the maximum value necessitates searching the entire set of data for each scatterer that is resolved. Images often require 100 or more scatterers for accurate representation and therefore the equivalent of 200 Gbytes of data must be retrieved from the storage media. On a small system, the data access and transfer times may be significant.

A GA may be used to locate multiple bright spots in one pass. A model may be formed from these bright spots and its effects subtracted from the original image as before. Further applications of the GA will locate any smaller points remaining. To calculate the image of the model, for a high-resolution image of 640 pixels on each axis, $3 \times (640^2) = 1,228,800$ Fast Fourier Transforms (FFT) need to be performed to create an image from the cross section data. On a typical desktop PC (Pentium 90Mhz), it would take around two hours to perform the transforms. To generate a model with 100 scatterers on a desktop PC

using this method would take over a week. This time scale and the massive storage requirement means that the iterative approach is not viable on a small system. Generating an image of one scatterer takes almost as long as generating an image of ten scatterers. Thus by processing multiple points in each pass of the data, vast savings can be made in image generation time. A multispecies GA that can identify seven or eight scatterers in each run will cut the processing time for a 100-scatterer model to around one day. This multimodal function approach can make model generation possible on a small system.

B. MultiModal Optimisation and Sharing

Most GAs use a single population of a single species. The algorithms are designed so the solutions represented by the different individuals converge on the single optimum solution of the objective function. In multimodal optimisation, the GA is designed to converge with multiple solutions. Each solution corresponds to a separate peak in the objective function.

There are a number of mechanisms that may be used to force a GA to exhibit multimodal behaviour.

1. **Iteration** – Many independent runs of the GA are performed in an attempt to identify all the peaks. This method is very inefficient as the larger peaks will often be found many times [6, Page 176].

2. **Sharing** – The sharing system operates by modifying the objective value that is seen by each individual. If a number of individuals all occupy the same peak in the objective function, they are made to share the objective value at that point [7]. This simple concept is enough to allow multiple stable populations to form.

3. **Crowding** – Crowding is a selective breeding technique where offspring are inserted into the population by replacing individuals that are genetically similar [8]. The process allows multiple stable populations to form.

4. **Sequential Niching** – The process operates by iterating the GA but maintains a record of the best solutions found. At each successive iteration of the GA, the peaks that correspond to the solutions found in previous runs are suppressed. This method is essentially a sequential version of the sharing process described earlier [9].

In order to identify multiple scatterers in each pass of the GA, either sharing or crowding methods must be used.

For the sharing process, a function that is related to the separation distance between two individuals' chromosomes is used to control the modification of the objective function. Eq. 4 (below) defines the sharing function used, with $d(\chi(i), \chi(j))$ defined as the distance between the chromosomes vectors $\chi(i)$ and $\chi(j)$, $s(i, j)$ is the sharing effect of i on individual j, and α, β are factors for modifying the function shape. When $\alpha = 1$, this function produces a linear variation that moves from unity at zero distance to $1 - \beta$ at a distance of ν and zero thereafter. If $\alpha \neq 1$, the function has an exponential form. Using a value of β less than unity has a similar effect to using high values of α, but without the processing overhead of the exponential calculations.

$$s(i, j) = \begin{cases} 1 - \left(\frac{d(\chi(i), \chi(j))}{\nu}\right)^{\alpha} \beta & d \leq \nu \\ 0 & d > \nu \end{cases} \quad (4)$$

$$\text{where} \quad d = d(\chi(i), \chi(j)) = |\chi(i) - \chi(j)|$$

For each individual, i, the distance is calculated from its chromosome to the chromosome of every other individual, j, in a population of N individuals and the values for each of the sharing functions are totalled (Eq. 5). The result is used to derate the image value at the point defined by the chromosome of i, $I(\chi(i))$ yielding a new objective value $\mathcal{O}(i)$. Eq. 6 shows the objective calculation $\mathcal{O}(i)$.

$$\mathcal{S}(i) = \sum_{j=1}^{N} s(i, j) \quad (5)$$

$$\mathcal{O}(i) = \frac{I(\chi(i))}{\mathcal{S}(i)} \quad (6)$$

These sharing functions work well, but for the large and complex optimisation surfaces found in scattering centre identification, large populations are required. Thus the requirement for every individual to be compared to every other produces a significant processing overhead. In an attempt to reduce the processing requirements, the sharing function has been modified to operate using multiple species rather than individual members [10]. This process combines the niche forming properties of the sharing process with the selective breeding of the crowding algorithm.

We can now define the position and spread of a species by the mean of the species chromosomes and their standard deviation. Eq. 7 and 8 define the spread and deviation, where n_k is the number of individuals in a species and $\chi(i, k)$ denotes the chromosome of individual i of species k. If we assume that the spread of individuals around the mean position is roughly Gaussian, a sphere with a two standard deviation radius from the mean will encompass the main bulk (74%) of the population. Thus we may define $2\sigma_k$ as representing the spatial distribution of population k.

$$\overline{\chi}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \chi(i, k) \quad (7)$$

$$\sigma_k^2 = \frac{1}{n_k} \sum_{i=1}^{n_k} |\chi(i, k) - \overline{\chi}_k|^2$$

$$= \frac{1}{n_k} \sum_{i=1}^{n_k} d(\chi(i, k), \overline{\chi}_k)^2 \quad (8)$$

Eq. 9 defines the modified sharing function, where ν_k is a sharing distance that varies with the spread, σ, of the species. The spread of the species is still limited to a minimum distance of ν. This sharing function is then applied to all N_s species, except the members' own, and the results summed (Eq. 10). The objective cost for the individual is

then derated by one plus the share value to account for the individual itself and is shown in Eq. 11.

$$s\left(\chi(i,j),\overline{\chi}_k\right) = \begin{cases} 1-\left(\frac{d(\chi(i,j),\overline{\chi}_k)}{\nu_k}\right)^2\beta & d \le \nu_k \\ 0 & d > \nu_k \end{cases} \quad (9)$$

$$\text{where} \quad \nu_k = \begin{cases} \sigma_k & \sigma_k > \nu \\ \nu & \sigma_k \le \nu \end{cases}$$

$$\mathcal{S}(\chi(i,j)) = \left.\sum_{k=1}^{N_s} \frac{n_k}{2\sigma_k} s(\chi(i,j),\overline{\chi}_k)\right|_{k \ne j} \quad (10)$$

$$\mathcal{O}(\chi(i,j)) = \frac{I(\chi(i,j))}{1+\mathcal{S}(\chi(i,j))} \quad (11)$$

The sharing function defined in Eq. 9 is based on a squared law rather than a linear function, i.e. with reference to Eq. 4, $\alpha = 2$. Individuals that are close to the species centre are affected more than those further away. An increase in calculation speed is gained by not requiring the square root of the magnitude of the distance to be taken. Unlike the fixed shape individual sharing functions used previously, the functions associated with each species are dynamic and vary with the geographical motion of the individuals within the species.

The ratio of the number of individuals in a species (n_k) to the species spread $(2\sigma_k)$ has been included as a factor in Eq. 5 to give Eq. 10. This ratio makes the influence that each species has on other individuals change dynamically with the species spread. Each species is now referred to by its mean position. When a species' population is widely dispersed $(\sigma \gg \nu)$, the function has little effect on other individuals. As a species' population converges $(\sigma \simeq \nu)$, the range of the function decreases but its influence increases. This added influence forces different species to separate as their populations converge. A minimum distance, ν, for the spread of the function is used to prevent different species from converging too closely to one another. This minimum distance helps increase the diversity of the geographical spread of the species.

C. Algorithm Construction

A real-valued chromosome with three parameters has been used to define each individual. The parameters are defined as being the (x, y, z) coordinates of a location in the three-dimensional ISAR image. The genotypic space, i.e. chromosome level, allows the parameters to have fractional components. In turn, the fractional component allows real-valued mutations to be applied. The genes are rounded to the nearest integer to obtain the phenotypic data that represent the picture element indices. The raw objective value is defined as the image intensity at the indexed point.

The algorithm follows the usual format of ranking, selection, crossover, mutation, and evaluation but with each species being processed separately. The same number of offspring as parents are generated and a total replacement policy is used. The total replacement policy helps to reduce the rate of convergence and allows the species to relocate themselves to minimise problems caused by overcrowding.

The fitness value $F(x)$ is assigned according to rank position p_x of individual x. The individual with the lowest $\mathcal{O}(x)$ (least fit) being assigned a rank position of 1 and the best individual being assigned rank position M. Eq. 12 details the calculation of $F(x)$.

$$F(x) = \left.\frac{2s(p_x-1)}{M-1} + (1-s)\right|_{0 < s \le 1} \quad (12)$$

where s is the *selective pressure* [6, Page 56] and may lie in the range $0 < s \le 1$. The emphasis that is placed upon the selection process may be controlled by adjusting s. A value of zero is never used as any bias between good and bad individuals is prevented and therefore no selection occurs. A value of unity gives the maximum selection where the chances of selecting the worst individual are near zero. The effect of reducing the selective pressure is to slow the convergence of the algorithm. In this algorithm, a selective pressure of 0.8 is used. This value has been selected empirically and allows the species to search the entire image but still converge satisfactorily.

Stochastic universal sampling [6, Page 57] is used to select M individuals from the population. This sampling process gives zero bias and minimum spread. The individuals selected are randomly shuffled and then paired up for breeding. Uniform crossover [6, Page 88] is used to generate two new offspring from each pair of parents.

This operator swaps individual gene-pairs between the parents with a probability of 0.5. For example, if we have two parents, a and b, both with chromosomes containing two genes, uniform crossover can be used to generate two offspring. The first offspring may have its first gene from parent-b and its second from parent-a. Due to the spectral spreading that occurs with the Fourier transforms in the ISAR image generation, each peak has long tails that spread out in the axis directions. Although a real-valued chromosome is used, the uniform crossover is suited to searching the image as individuals often settle onto a tail emanating from a peak. If the two parents are each lying on different tails of the same peak, after crossover, the offspring may lie *exactly* on the peak. Other recombination techniques that create offspring by combining genes proportionally are unlikely to score a direct hit on the peak.

Genes are mutated with a probability of 0.3. This probability will mutate, on average, approximately one gene per chromosome. The range of the mutation is governed by Eq. 13, where G is the generation number and G_m is the maximum number of generations. This *nonuniform* mutation [6, Chapter 6] function is unity initially and progresses to zero at the final generation and is used to modify the maximum deviation from the current gene value. Initially, the gene can mutate to any value within its range but this range is reduced with time. The range modifier function forces the GA to converge on a solution by confining the offspring of each subsequent generation to a diminishing region. In the final stages of the GA, the mutation range is limited to a very small locality, forcing the species to converge on the true local optimum. The function shape allows the GA to perform a thorough search in the early

generations but still retain the convergence properties of nonuniform mutation toward the end phase of the algorithm. The use of a real-valued chromosome in this GA is due to the use of this specific nonuniform mutation operator. Figure 3 depicts the function shape graphically.

$$R(G) = 1 - \left(\frac{1 - \cos\left(\frac{(G-1)}{(G_m-1)}\pi\right)}{2} \right)^2 \qquad (13)$$
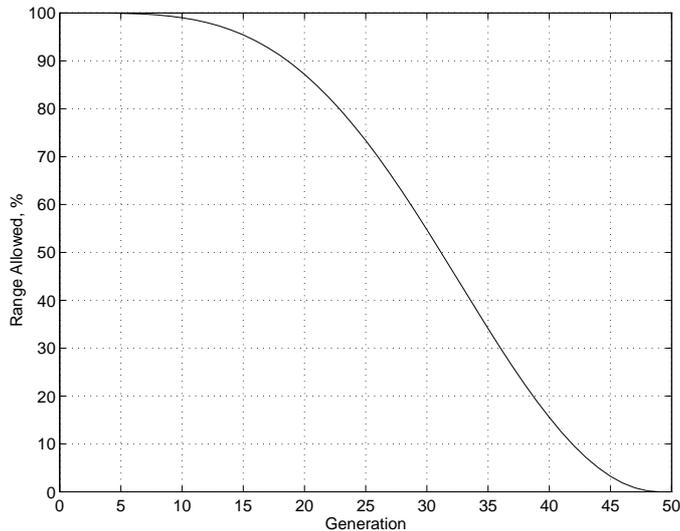

Fig. 3. Nonuniform mutation range modifier function used in GA-1

The objective function for the offspring is calculated as detailed previously, based on the statistics of the parent population. Policies other than total replacement have been tried but appear to offer little benefit as the objective function is effectively dynamic with the motion of the species. The use of dual chromosomes (diploidy) and a dominance mechanism have been tried to improve the search process by allowing species to develop a memory of good peaks they have occupied in the past. The effect was to slow the convergence of the algorithm but no improvements in the results were noticed. The lack of improvement was probably due to the low number of generations used in the algorithm.

The algorithm is terminated after 50 generations and the best overall individual of each species is recorded as a peak location. The small number of generations combined with forced convergence has been chosen to give a short and consistent execution time. As the algorithm is being applied in an iterative fashion, the requirement is to identify any of the peaks in the image, rather than the highest set. Therefore the algorithm's ability to identify the global peak in each run is sacrificed for speed. The use of 50 generations has been chosen empirically to allow the population to begin to converge before the non-uniform mutation operator begins to have a significant effect.

Once the algorithm has terminated, duplicate peaks and any that are within one spread distance ($\leq \nu$) are removed.

A range of species is used, each with a different population size. This range of species sizes introduces a slight bias into the algorithm where the smaller species are able to move more rapidly than the larger species but have a weaker hold on any peaks they find. The larger species move slowly but are capable of evicting small species from peaks that are already colonised.

Comparing the multispecies GA to existing approaches that search for the highest peak, if we have an algorithm with 750 individuals and run it for 50 generations, it will require 37500 objective calculations. The algorithm can locate as many peaks as there are species, although a 70% identification rate is more realistic. If we apply the algorithm 10 times to identify 100 scatterers in a high resolution image, a total of 375,000 accesses are required to the image data. The conventional iterative model conversion approach accesses nearly 560,000 times as much image data and generates ten times as many ISAR images to achieve the same model resolution.

IV. Model Tuning

A. Introduction

With low-resolution images, the scatterer locations generated by the first GA may be a significant distance away from the optimum positions. Even with high-resolution data, if two scatterers are very close, the image peak positions may not be truly aligned with the actual scatterer location [11]. By fine tuning the model, these errors can be reduced. The fine tuning process will ultimately result in the model requiring fewer scatterers to match the target data and so reduce the burden on the third GA.

B. Image Registration

The first stage in converting the scatterer image details into a point scatterer model is to register the image with the model. The image registration is achieved by placing a scatterer at the origin of the model and generating an ISAR image of it. The highest point in the image will correspond to the scatterer in the model, giving the image–model zero location ±1 resolution cell. A rough amplitude scaling can be calculated from the brightness of the peak in the image. Knowing the true image resolution from Eq. 2 and 3, the model position of a point that is a fixed distance on each axis away from the centre is calculated. The distance chosen must be related to the position of the centre point so that the peak should not fall outside the image region. If it does lie outside, an aliased image will be present, but at a false location. This second peak allows the image scaling to be be verified and any scale inversions identified. It is possible for increasing x in the model space to lead to decreasing x in the image etc. This indicates that the data ordering has been reversed in some way and must be corrected. The image–model registration only needs to be performed once at the beginning of the conversion process.

C. Model Generation

Once the zero location and scale have been verified, the location of each identified scatterer in the image can be transformed directly into a point in the model, with an accuracy of ±1 resolution cell. If the images are high resolution, for example greater than 512 samples in each axis, the resolution induced error will typically be of the order of a few centimetres or so, and therefore close enough for conversion purposes. The amplitude scale factor derived from the zero registration should also be accurate enough for direct conversion. An image of the model can then be subtracted from the original ISAR image to reveal the smaller scatterers that have not yet been located.

If the image is of low to medium resolution, i.e. 32 to 512 samples, the error in the scatterer's location becomes progressively worse and may ultimately be a few metres. Experience has shown that the observed peak amplitude in these cases may range from near zero, up to the true peak value. To cope with these images, the raw model positions must be fine-tuned in an attempt to reduce positional and amplitude errors. Figure 4 demonstrates the effects of a small number of samples on the image of a scatterer when the scatterer does not align with the sample position. When the peak of the image coincides with the sample location, the indicated amplitude is accurate. As the sample point is shifted left or right, the measurement error increases, with the indicated peak height being less than the actual peak.
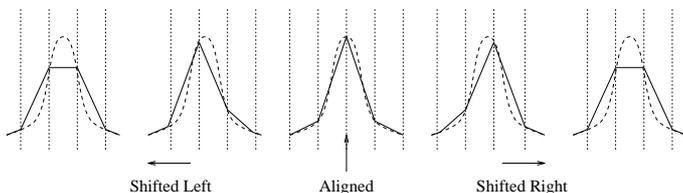

Fig. 4. Effects of mis-alignment between scatterer and sample instant

D. Genetic Algorithm Operation

The following GA is designed to adjust the raw scatterer positions to improve the match to the required image. It should be noted that as the image resolution decreases, the work of this GA increases and the load on the first GA for scatterer location is reduced. This shift in processing load should be adjusted by the designer for each different image type that is to be processed.

To reduce processing overheads, instead of calculating and comparing three-dimensional images, three one-dimensional images are used [4]. These images are formed by taking data from the three principle axes of the RCS pattern and using a Fourier transform to convert the RCS data to range information. Typically, if the RCS data in the region around the centre of the azimuth, elevation, and frequency bands are of most interest, the RCS would be measured at the mean azimuth and mean elevation and over the full frequency sweep, then at the mean elevation

and mean frequency with a full azimuth sweep etc. If we have a $64 \times 64 \times 64$ sample image, for full conversion, 12,288 Fourier transforms are required. Only three are required if the one-dimensional approach is used. This major reduction in the processing overhead is offset by reduction in the fidelity of the error measurements. It has been observed that the fine tune operation is not compromised by the use of a reduced set of data. If very low-resolution data are used, the processing of the full image is not too severe. The use of the full image in the comparison process may be justified to increase the fine tuning capabilities of the GA.

If the tuning process is not perfect, position and amplitude errors in the model will lead to errors in the image. As the model image is subtracted from the required image, any peaks in the model that are smaller than they should be will leave a positive residue peak in the image after subtraction. This smaller scatterer will be identified in subsequent passes and reduced further. This mode of operation eventually leads to models with an excess of scatterers. If the scatterers in the model are larger than they should be or in the wrong position, a negative result is obtained in these areas of the image after the subtraction process. This error cannot be corrected in subsequent passes of the algorithm and causes bright spots in the ISAR image that are too large. The negative error is highly undesirable and is a problem with any technique that operates by an iterative scatterer subtraction process.

The problem is addressed in the GA by calculating the amount of overshoot (negative error) and undershoot (positive error) of the fit separately. A multiobjective approach is used that allows the designer to trade between the image errors. A slow fitting process may be used that minimises the undesirable overshoot errors but leads to larger models. Alternatively, a less stringent fitting scheme may be applied that minimises undershoot and therefore uses less scatterers, but at the expense of ISAR image accuracy. Alternatively a compromise can be drawn between the two objectives that attempts to minimise the negative error problems without creating a large model.

E. Non-dominated Ranking

A Pareto optimal set of results [3, Pages 197–201] may be formed where no single solution is better than any other in both objectives. These solutions are said to be *non-dominated* as no solution can be chosen in preference to the others based on the two objectives alone. There exists a single Pareto optimal set of solutions to the problem. At any intermediate stage of optimisation, a Pareto set of results will have been identified. This set may or may not be the optimal set.

A non-dominated ranking method [12] is used in the GA to generate and maintain a Pareto set of results. Conventional GAs often use a ranking method where the calculated objective values are sorted and assigned a rank that is dependent only upon their position in the list, rather than their objective value. The ranking operation helps to prevent premature convergence of the GA.

The non-dominated ranking system operates by first identifying the non-dominated solutions in the population and assigning them a rank of one. A dummy value (1 in this implementation) is assigned to these solutions and the sharing process detailed in section III-B is applied. With the sharing, the dummy values of the individuals' are reduced if they have near neighbours (on a chromosome level). The sharing process ensures that a spread of solutions is obtained across the Pareto front. The minimum value assigned to the level one solutions is identified and then reduced slightly (by 1%). The level one individuals are removed from the population and the identification–sharing process repeated on the remaining set, using the reduced dummy value for the sharing operation. The ranking process is continued until all of the individuals have been accounted for. The resulting objectives are intended to be used with a *maximisation* strategy. Figure 5 illustrates the algorithm.
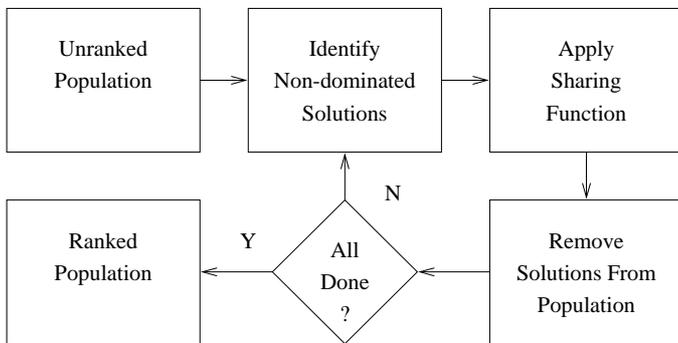

Fig. 5. Non-dominated ranking process

The conventional fitness and selection processes are then applied as normal to the objective obtained by the non-dominated ranking and sharing operation. An elitist strategy is developed that preserves an entire Pareto front of P solutions from generation to generation. To maintain a working population of N individuals, the Pareto set from the previous generation is concatenated with the working population and then N offspring are generated from the $N + P$ parents. After evaluation, the offspring become the new working set of individuals. The new Pareto set is calculated from the population of solutions that results from concatenating the new working set of individuals and the old Pareto set, thus choosing the best from the new solutions and old Pareto front. The number of solutions that comprise the Pareto front, P, is dynamic. Kumar and Rockett [13] discuss procedures that may be used if assurances are required that the true Pareto optimal set of solutions has been identified.

F. Algorithm Construction

A real-valued chromosome is used, held in a matrix structure that has four columns corresponding to $[\Delta \mathbf{x} \; \Delta \mathbf{y} \; \Delta \mathbf{z} \; \mathbf{a}]$, where $\Delta \mathbf{x}$, $\Delta \mathbf{y}$ and $\Delta \mathbf{z}$ are offsets from the raw scatterer position and $\mathbf{a}$ is the amplitude. The chromosome matrix has the same number of rows as the number of scatterers

identified in the scatterer location algorithm. The positional offsets are limited to ± 1.25 resolution steps and the amplitude is allowed to range from zero to 50% larger than the largest identified scatterer. The ranges have been chosen empirically to allow the GA the flexibility to explore extreme solutions whilst minimising the execution time. In the phenotypic (i.e. model) space, the scatterers corresponding to each chromosome are concatenated to the previously identified model before the images are generated. Trials indicate that for a 3D image with a resolution of 64 samples on each axis, using the algorithm with around 100 individuals and running for 500 generations will give good results.

Eq. 14 and 15 show the two objectives that are used in the fitting process, where $E(x, k)$, defined in Eq 16, is the error between the required image, $I(k)$, at point k and the image of the model, $M(x, k)$, for solution x; N is the number of points in the image; $\mathcal{O}_1(x)$ is objective one and is a measure of mean squared overshoot; $\mathcal{O}_2(x)$ is objective two and describes undershoot. The objectives are both to be minimised to establish the Pareto optimal front.

$$\mathcal{O}_1(x) = \frac{1}{N}\sum_{i=1}^{N}\left\{\begin{array}{ll} E(x,i)^2 & E(x,i) > 0 \\ 0 & E(x,i) < 0 \end{array}\right. \quad (14)$$

$$\mathcal{O}_2(x) = \frac{1}{N}\sum_{i=1}^{N}\left\{\begin{array}{ll} 0 & E(x,i) > 0 \\ E(x,i)^2 & E(x,i) < 0 \end{array}\right. \quad (15)$$

$$E(x,k) = I(k) - M(x,k)|_{k=1\ldots N} \quad (16)$$

The fitness function as defined in Eq. 12 is applied but with a selective pressure of $s = 1$. This selective pressure gives the maximum bias towards the most fit solutions. Stochastic universal sampling is used to select N individuals from the $N + P$ set described previously. Uniform crossover is applied where parts of the paired chromosomes are exchanged. A mutation rate of 0.25 is applied along with nonuniform mutation as described previously in section III-C. This process increases the chances that the solutions found will lie exactly at optimum positions.

At the end of the algorithm, one solution is chosen from the Pareto set. Which solution is chosen is determined by the design strategy that is being employed. A slow but accurate method will chose the solution where $\mathcal{O}_1$ is the smallest (least overshoot) while a less stringent strategy will pick the solution that minimises $\mathcal{O}_2$ (least undershoot). An average solution may be obtained by minimising the sum of the normalised objective terms. For each objective, the minimum and maximum limiting values are found from those individuals in the Pareto set. These limits are used to normalise the objective values to lie in the range zero to one. The normalised values are then summed for each individual. The individual which has the lowest sum is chosen.

V. FITTING CYCLE TERMINATION

A. Introduction

After each fine tuning phase, the model is tested to establish if enough scatterers have been identified to allow

the fitting cycle to end and the reduction phase to begin. The test is applied to the best RCS pattern that can be fitted to the current model. If the required cross section pattern matches sufficiently the best pattern that can be fitted to the model, the fitting cycle may be terminated.

B. Model Testing

Scatterer amplitudes and phases are fitted using a constrained least squares process [14] in an attempt to match the required RCS pattern. The Kolmogorov–Smirnov statistical test is applied to establish the accuracy of the model. This test gives a measure of statistical similarity that is independent of the amount and mean amplitude of the RCS data used in the comparisons. If the RCS of the model does not fit the required data, an image is generated from the model and this image is subtracted from the original image. This process removes the scatterers that have been identified. The new image is then passed back to the first GA to identify a new set of peaks.

C. Constrained Least Squares Fitting

If the field pattern described by the N target RCS data are denoted by a vector g_o and the M model amplitudes and phases are described by a vector f, Eq. 17 describes a linear operator T that relates the two.

$$[T]_{N \times M}[f]_{M \times 1} \approx [g_o]_{N \times 1} \qquad (17)$$

Eq. 18 defines the standard unconstrained least squares solution, where T^* indicates the complex conjugate and T^T indicates matrix transpose.

$$f = [[T^*]^T T]^{-1} [T^*]^T g_o \qquad (18)$$

Unfortunately, the simple least squares method can give wild overestimates for scatterer magnitude values in the model. A better method is to use a constrained least squares approach [14].

If Φ_i and λ_i denote the ith Eigen vector and Eigen value of the matrix $[T^*]^T T$, ie.

$$[[T^*]^T T]\Phi_i = \lambda_i \Phi_i$$

Then the constrained least squares approximation of f is defined by Eq. 19.

$$f = \sum_{i=1}^{M} \frac{c_i}{(\lambda_i + \alpha)} \Phi_i \qquad (19)$$

where

$$c_i = [\Phi^*]^T [T^*]^T g_o$$

The value α may be found using the Newton–Raphson iterative method, Eq. 20 details the calculation; where C is the applied constraint. A starting value of $\alpha_0 = 1$ is suggested.

$$\alpha_1 = \alpha_0 - \left(\frac{C - \sum_{i=1}^{M} \frac{|c_i|^2}{(\lambda_i + \alpha_0)^2}}{2 \sum_{i=1}^{M} \frac{|c_i|^2}{(\lambda_i + \alpha_0)^3}} \right) \qquad (20)$$

The square of the norm of f defined in Eq. 21 is limited by the value of the constraint C.

$$||f||^2 = \sum_{i=1}^{M} |f_i|^2 \qquad (21)$$

The required constraining value for $||f||^2$, ie. C, is found by calculating the squared norm of the image. Eq. 21 is applied to the image with f_i representing each picture element. As the image intensity is determined by scatterer amplitudes, the squared norm should be approximately the same as the squared norm of the model.

D. Statistical Testing

The Kolmogorov-Smirnov (K–S) statistical test [15, Pages 472–475] is used to compare the RCS of the fitted model with the target RCS. The test gives a figure of merit for the similarity between the cumulative distribution functions of the RCS probability distributions. The K–S number, λ_{KS}, may be related to a probability that the sets of data are drawn from the same distribution.

If we have two cumulative distribution functions S_1 and S_2 of size N_1 and N_2 respectively, the K–S statistic is

$$\lambda_{KS} = \sqrt{\frac{N_1 N_2}{N_1 + N_2}} \left(\max_{-\infty < x < \infty} |S_1(x) - S_2(x)| \right)$$

In the case of the null hypothesis 'sets of data drawn from the same distribution', the distribution of the K–S statistic can be calculated giving the significance of any observed non-zero value of λ_{KS}.

The significance may be calculated using

$$Q_{KS}(\lambda) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2 \lambda^2}$$

which is monotonic with limiting values $Q_{KS}(0) = 1$ and $Q_{KS}(\infty) = 0$. In terms of this function, the P-value ($\hat{\alpha}$) of an observed value of λ_{KS} is given approximately by

$$\text{Prob}(\lambda_{KS} > \text{observed}) = Q_{KS}(\lambda_{KS})$$

The approximation becomes asymptotically accurate as N becomes large. Typically $N > 20$ is acceptable.

VI. Model Reduction

A. Introduction

The process of model fitting can yield models with large numbers of scatterers. This large amount of model data can create extended simulation times. If we accept that a measured or calculated RCS will never be a perfect representation of the real target [16], [17], small degradations in data fidelity are acceptable. Therefore, if we remove some of the scatterers in an n-point model, we should be able to readjust the model to give an approximation to the desired RCS. As the model is used tens of thousands of times in a typical engagement, *any* reduction in model size is beneficial.

As RCS is governed by scatterer interactions, simple iterative reduction methods that remove the points that appear to have the least influence may not always select the best combination of scatterers [18]. A small scatterer may have little effect on its own but may be dominant when paired with another similar scatterer. An exhaustive search of all possible model combinations in an attempt to find an optimum solution is often impractical.

B. Algorithm Construction

A n-bit chromosome has been used to define the model structure, $[s_1 \, s_2 \, \ldots \, s_n]$, where n is the number of scatterers in the model. Each bit corresponds to a scatterer. If a bit is '1', the corresponding scatterer is present in the model; if it is '0', the scatterer is omitted. Scatterer locations are kept fixed, magnitudes and phases are fitted using the least-squares fitting method each time a new individual is created.

The performance of each individual is calculated by first fitting weightings to the selected scatterers using the constrained least-squares method defined in section V-C and then generating the N radar echo data samples for the region of optimisation. The mean squared error of the radar echo from the individual's model compared with the original required radar echo is calculated at each stage by applying Eq. 22, allowing the effects of the reduction to be monitored for the current region of optimisation. In Eq. 22, $r(x)_i$ is the radar echo of model x at point i, g_i is the required radar echo and N is the number of data samples. The number of active scatterers in the model is calculated using Eq. 23; where $\chi(x)_i$ is gene i in the chromosome of individual x. The number of scatterers is used along with the radar echo error to generate a Pareto set of results where model size is traded against reproduction accuracy.

$$\mathcal{O}_1(x) \quad = \quad \frac{1}{N} \sum_{i=1}^{N} |r(x)_i - g_i|^2 \qquad (22)$$

$$\mathcal{O}_2(x) \quad = \quad \sum_{i=1}^{n} \chi(x)_i \qquad (23)$$

The non-dominated ranking methods described in section IV-E are used to maintain the Pareto population. A selective pressure of $s = 1$ is used to give maximum distinction between good and bad solutions. Multipoint crossover [19, Page 13] is used to generate two new offspring from each pair of parents with a crossover rate of 0.8. This value has been chosen empirically to increase the rate of convergence in the initial stages of the algorithm. A mutation rate of $1/n$ is used, where n is the number of scatterers. This should mutate on average one gene per chromosome.

After each run is terminated, the Pareto set is recorded as the solution. If an exhaustive search is performed, 2^n objective calculations are required. If we take the GA as requiring 100 individuals and 500 generations, 50,000 calculations will be performed. This means that a model of 15 scatterers or less is best reduced using an exhaustive search.

VII. EXAMPLE RESULTS

A. Introduction

Two example model fitting trials are presented in this section, one on measured two-dimensional data and one on a simulated three-dimensional image. The examples are typical of many processed models and demonstrate the versatility of the GA approach with its ability to process many different types of image.

The strategy used for fine tuning in both trials was to try to find an average solution. In both cases, the RCS data used in the fitting process was derived from the same source as the 1D ISAR images described previously. These azimuth, elevation (3D only), and frequency traces are concatenated to form a single pattern to match. This gives general coverage of the cross section data whilst minimising computation time. Table I summarises the structure of the three GAs used in the model generation process.

B. Two-Dimensional Image

The two-dimensional data were measured from a real target at zero elevation and with the image conditions specified in Table II; where range resolution and total range are in metres. The results were obtained with the GAs operating under the conditions shown in Table III.

TABLE II
CONDITIONS FOR ISAR IMAGE GENERATION (2D)

Range	Res.	Total	Steps	Sweep Range
Slant	0.2986	76.44	256	2.5GHz : 3GHz
Cross	0.2963	18.96	64	$-5.1°$: $5.4°$

TABLE III
OPERATING CONDITIONS FOR GAs (2D)

Parameter	GA-1	GA-2	GA-3
Maximum generations	50	100	300
Total no. individuals	100	20	25
No. of species	7	1	1
Selective pressure	0.8	1.0	1.0
Crossover rate	1.0	1.0	0.8
Mutation rate	0.3	0.25	$1/n$
Minimum share dist., ν	3	10	1
Share shaping, β	0.5	1.0	1.0

The first fitting stage comprising GA-1 and GA-2 required 38 iterations to locate 174 scatterers with RCS K–S significance of 90%. Figure 6 shows the locations of the scatterers and Figure 7 shows the original ISAR image. It is quite apparent that the scatterer locations closely follow the form of the ISAR image.

The third GA was then used to generate a set of smaller models. Figure 8 shows the set produced after 300 generations of GA-3. The cost function used is as defined in Eq. 22. Although the cost function is a good means of

TABLE I
SUMMARY OF GA STRUCTURES

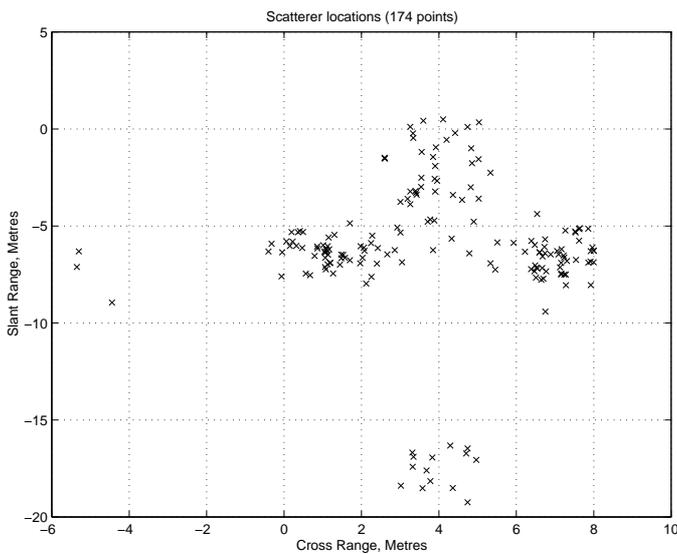| Feature	GA-1	GA-2	GA-3
Type	real	real	binary
Chromosome	$[x\ y\ z]$	$[\Delta\mathbf{x}\ \Delta\mathbf{y}\ \Delta\mathbf{z}\ \mathbf{a}]$	$[s_1\ s_2\ \dots\ s_n]$
Multispecies	yes	no	no
Non-dominated ranking	no	yes	yes
Non-uniform mutation	yes	yes	no
Objective	Image value (Eq. 11)	Overshoot (Eq. 14) Undershoot (Eq. 15)	Mean sq. error (Eq. 22) Model complexity (Eq. 23)

Fig. 6. Scatterer locations identified from the 2D ISAR image

Fig. 7. Original 2D ISAR image

quantifying the error between the model and the required RCS, it is difficult to gauge the optimum model size to use. Figure 9 shows the the results of the K–S statistic when applied to the reduction set. It is clear that the best identified model has 128 scatterers as the 90% significance level is maintained. Allowing GA-3 to run for more generations would eventually provide a smooth K–S curve but may not improve on the model size. The option to terminate the GA early is left to the designer.

Fig. 8. RCS cost against number of scatterers for the 2D data

Figure 10 shows the scatterer locations in the reduced model. The RCS of the model (solid) compared to the required cross section (dashed) is shown in Figure 11. The ISAR image of the model is shown in Figure 12

C. Three-Dimensional Image

The three-dimensional test data were generated from a semirandom model consisting of 50 scatterers roughly shaped to mimic an aircraft and with the image conditions specified in Table IV, where range resolution and total range are in metres. The following results were obtained with the GAs operating under the conditions shown in Table V.
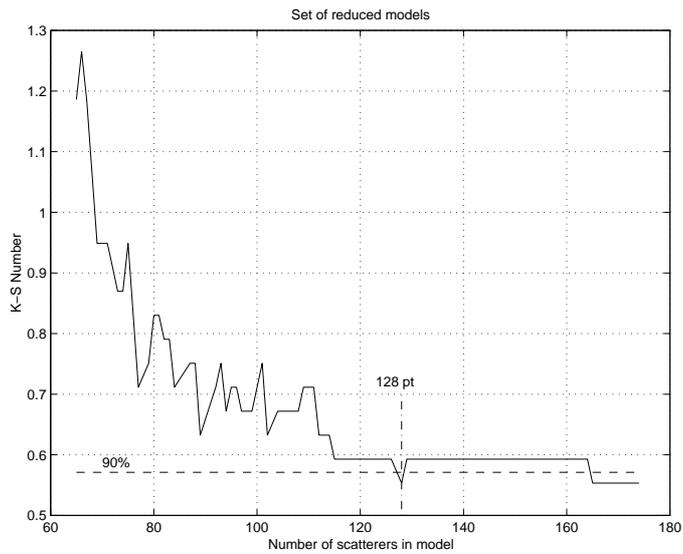
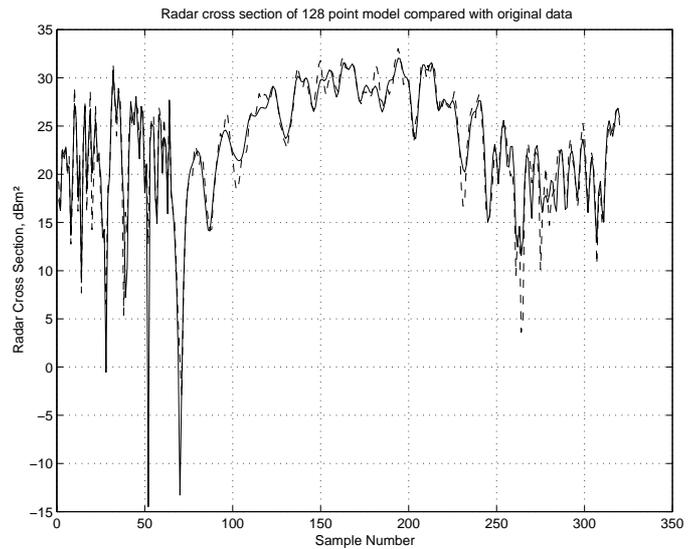Fig. 9. K–S significance of reduced models (2D)
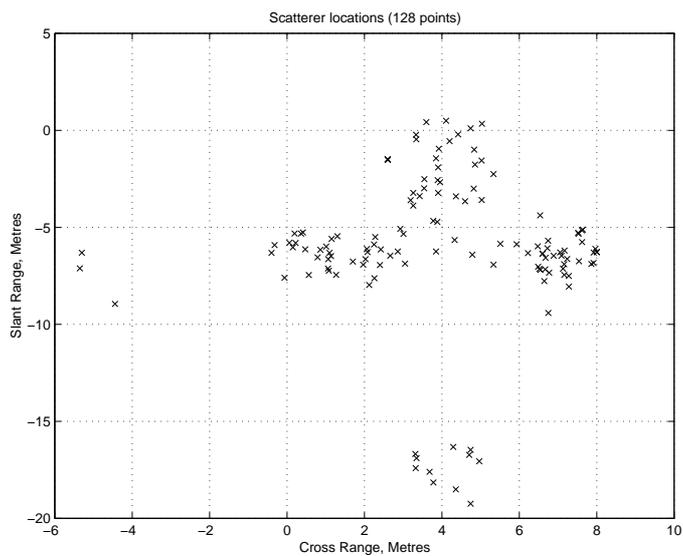

Fig. 11. Radar cross section of 2D reduced model
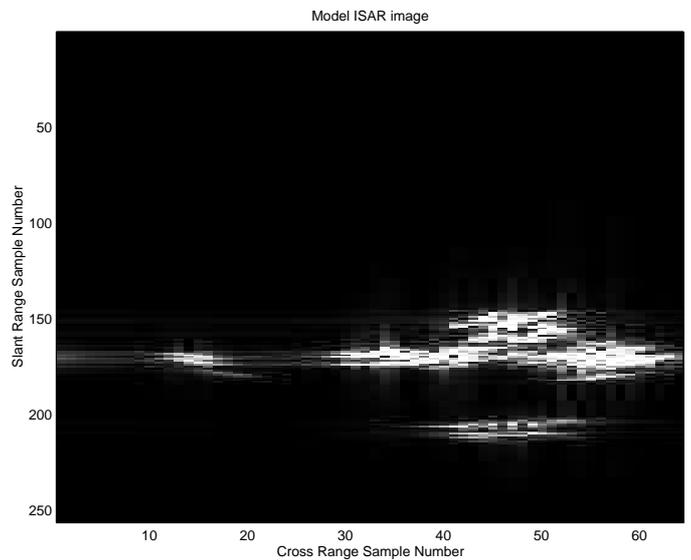

Fig. 10. Scatterer locations of 2D reduced model

Fig. 12. ISAR image of 2D reduced model

TABLE IV
Conditions for ISAR Image Generation (3D)

Range	Res.	Total	Steps	Sweep Range
Slant	0.3845	24.61	64	10.8 : 11.2 GHz
Cross, Az	0.3906	25.00	64	$-1°$: $1°$
Cross, El	0.3906	25.00	64	$-1°$: $1°$

TABLE V
Operating conditions for GAs (3D)

Parameter	GA-1	GA-2	GA-3
Maximum generations	50	500	500
Total no. individuals	750	100	150
No. of Species	10	1	1
Selective pressure	0.8	1.0	1.0
Crossover rate	1.0	1.0	0.8
Mutation rate	0.3	0.25	$1/n$
Minimum share dist., ν	3	10	1
Share shaping, β	0.5	1.0	1.0

The first fitting cycle required 24 iterations to identify 173 scatterers that gave a fitted RCS to a K–S significance level of 95%. The third GA was then applied to obtain a set of reduced models. Figure 13 shows the set produced after 500 generations of GA-3. Figure 14 shows the the results of the K–S statistic when applied to the reduction set. It is clear that the best identified model has 88 scatterers as this gives an acceptable 90% significance level. Again it is up to the designer to decide how long the reduction algorithm should be run for. The smallest model of suitable fidelity should always be used to minimise engagement simulation times. Figure 15 shows the scatterer locations in the reduced model. The RCS of the model (solid) compared to the required cross section (dashed) is shown in Figure 16.

Fig. 14. K–S significance of reduced models (3D)

Fig. 13. RCS cost against number of scatterers for the 3D reduction phase

Fig. 15. Scatterer locations of 3D reduced model

D. Discussion

Despite the ISAR images having a spatial resolution of approximately 30 centimetres, models were generated that have a reasonable number of scatterers and still approximate the targets RCS. The examples demonstrate clearly the ability of the multiple GA method to process both low-to-medium resolution data and 2D/3D data. The traditional iterative approach requires data with ten times the resolution to produce a good model. The multiple algorithm approach may be applied equally to radar cross section data generated from CAD models, where many numerical short-cuts may be applied to increase the performance further [4].

VIII. Compound Model Formation

The models are combined using a *binary space partition (BSP) Tree* structure [20, Pages 675–680] to allow the correct point scatterer model to be rapidly retrieved for any aspect angle. The structure allows the models generated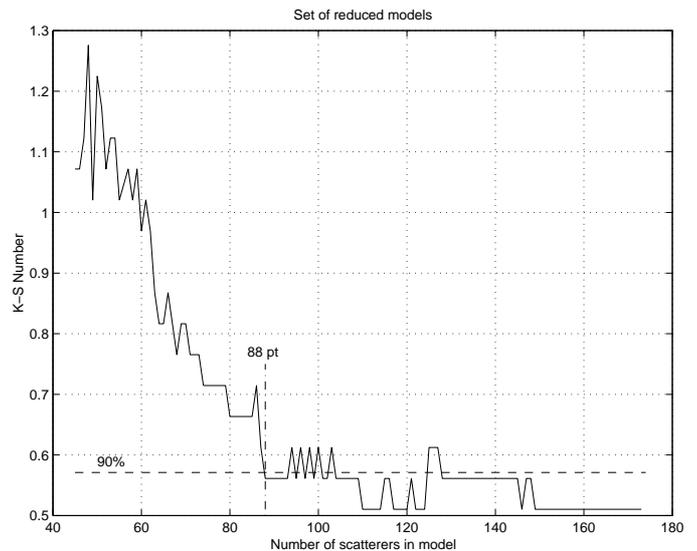 for small aspect angles and frequency ranges to be combined to cover a larger region of interest. Azimuth and elevation are normally the main decision variables used to generate the tree but models that vary with frequency, range and polarisation for example can be incorporated easily.

Space partition trees are designed to split an object into its component parts in a manner that makes them easily retrievable. If an object is split into N components, on average $\log_2(N)$ tests must be performed to establish the correct component part to be used. For example, if an object has 1000 component parts, an average of just under 10 tests must be performed. If we have the example object space shown in figure 17, the components *1*, *2* and *3* can be separated by the two lines *a* and *b*. The test procedure will be to determine whether the viewing position lies either to the front or reverse side of the partitioning line under consideration. Although Figure 17 shows
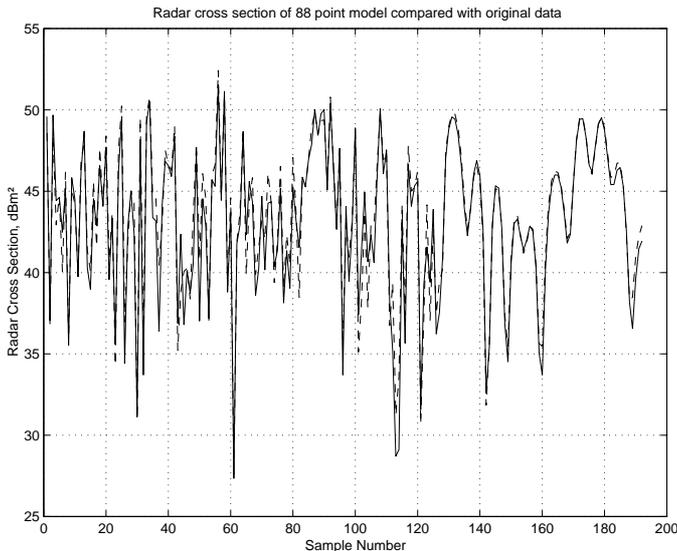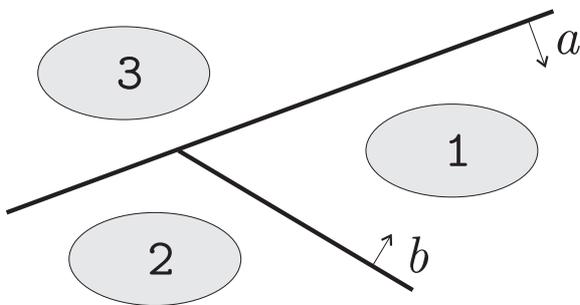
Radar cross section of 88 point model compared with original data

Fig. 16. RCS of 3D reduced model

Fig. 17. Space partitioned model

Fig. 18. *BSP Tree* for Figure 17

IX. Conclusions

The work presented in this paper demonstrates that sometimes the application of many small GAs may be preferable to using one large and complex one. Attempts to solve this model identification problem with one large algorithm have proved fruitless. The multiple algorithm approach is robust and will provide repeatedly a solution to the problem; even though some of the algorithms are forced to converge, thereby limiting their potential. The forced convergence gives more uniform operation with respect to job execution times.

The method allows both two-dimensional and three-dimensional images of low, medium, or high resolution to be processed. The multiple algorithm approach gives the model designer more flexibility in applying constraints than would be available from a single GA. The binary space partition tree method for constructing compound models allows scatterer models with different complexities and coverage to be accessed efficiently. The multiple algorithm approach requires far fewer calculations than the traditional iterative method and makes model generation viable on a small system.

the partitioning of a two dimensional space, the *BSP Tree* technique will extend to *n-dimensional* space, where the sub-spaces are divided by a structure of $n-1$ dimensions. Therefore a three-dimensional world will be partitioned by two-dimensional structures, i.e., planes.

The tree is constructed by recursively splitting the model into *sub-spaces* using partition lines defined by the junctions between the models. Components which lie on the front side of the partitioning line are placed in the *left* branch of the tree. The components in the remaining sub-space are placed in the *right* branch. Figure 18 shows the complete tree for the components in Fgure 17.

In a typical model, we can define azimuth as running from $-180°$ to $+180°$ and elevation from $-90°$ to $+90°$. We may then have a partitioning line defined as being, say, $5°$ in azimuth. If the *front* side of the line is defined as being *increasing* angle, the models may be split about this line forming the first division of the tree. Each branch is then further sub-divided until there is only a single model at each node. During run time, the first test to be made will be to check if the missile position relative to the target is greater than $5°$. If it is then the left branch of the tree is traversed, else traverse the right side.
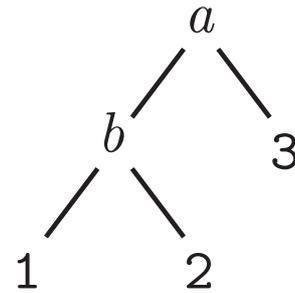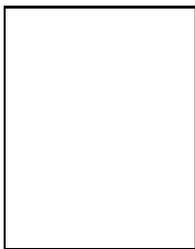
References

[1] D. R. Wehner, *HIGH RESOLUTION RADAR*, Artech House, 1987.
[2] M. J. Byrd, B. D. Jersak, B. D. Krenek, and A. J. Blanchard, "Demonstration of 3D microwave holography," in *IGARSS '94 International Geoscience and Remote Sensing Symposium*, Pasadena, CA, USA, 8–12 August 1994, pp. 918–920, IEEE.
[3] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.
[4] R. Bhalla and H. Ling, "Three-dimensional scattering centre extraction using the shooting and bouncing ray technique," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 11, pp. 1445–1453, Nov. 1996.
[5] C. L. Breuille and E. Caille, "Modeles multi-points brillants de signatures radar de cibles," in *Radar International Conference*, Paris, France, 3–6 May 1994, Soc. Electr. & Electron, pp. 325–329.
[6] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer–Verlag, 2nd edition, 1992.
[7] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Second In-*

ternational Conference on Genetic Algorithms: Genetic Algorithms and Their Applications, 1987, pp. 41–49.

[8] S. W. Mahfoud, "A comparison of parallel and sequential niching methods," in *Sixth International Conference on Genetic Algorithms*, San Mateo, CA, USA, 1995, pp. 136–143, Morgan Kaufmann.

[9] D. Beasley, D. R. Bull, and R. R. Martin, "A sequential niche technique for multimodal function optimization," *Evolutionary Computation*, vol. 1, no. 2, pp. 101–125, 1993.

[10] E. J. Hughes, M. Leyland, and B. A. White, "A multi-species genetic algorithm applied to radar scattering centre identification in three-dimensions," in *GALESIA '97 Conference*, Glasgow, UK, 1–4 September 1997, IEE Pub. No. 446, pp. 472–477.

[11] A. W. Rihaczek, "Radar resolution of ideal point scatterers," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 2, pp. 842–845, Apr. 1996.

[12] N. Srinivas and K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1995.

[13] R. Kumar and P. Rockett, "Assessing the convergence of rank-based multiobjective genetic algorithms," in *Genetic Algorithms in Engineering Systems: Innovations and Applications. (GALESIA '97)*, Glasgow, 2–4 September 1997, pp. 19–23, IEE Conference Publication No. 446.

[14] J. R. Mautz and R. F. Harrington, "Computational methods for antenna pattern synthesis," *IEEE Transactions on Antennas and Propagation*, vol. 23, no. 7, pp. 507–512, July 1975.

[15] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *NUMERICAL RECIPES The Art Of Scientific Computing (FORTRAN Version)*, Cambridge University Press, 1989.

[16] S. D. Turner, "RESPECT: Rapid electromagnetic scattering predictor for extremely complex targets.," *IEE Proceedings, Part F*, vol. 137, no. 4, pp. 214–220, Aug. 1990.

[17] J. M. Williams, "Radar cross-section of large structures with complex microgeometry.," *IEE Proceedings, Part F*, vol. 137, no. 4, pp. 221–228, Aug. 1990.

[18] E. J. Hughes and M. Leyland, "Radar cross section model optimisation using genetic algorithms," in *RADAR '97 Conference*, Edinburgh, UK, 14–16 October 1997, IEE Pub. No. 449, pp. 458–462.

[19] A. M. S. Zalzala and P. J. Flemming, Eds., *Genetic algorithms in engineering systems*, The Institution of Electrical Engineers, 1997.

[20] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*, Addison-Wesley, 2nd edition, 1990.

Maurice Leyland was born in Lancashire, UK, in 1935. He graduated from the University of Birmingham, England, in 1958 with a degree in physics. He served in the Royal Air Force, spending most of his time at the Royal Air Force College, Cranwell. While in the Royal Air Force he gained an MSc in Electronics from Southampton University, England. On leaving the Royal Air Force he joined Cranfield University at the Royal Military College of Science, where he lectures primarily on radar. He is a Chartered Engineer and a member of the Institute of Electrical Engineers.

Evan J. Hughes was born in Northamptonshire, UK, in 1971. He received his BEng and MEng degrees in Electrical and Electronic Engineering from the University of Bradford, England, in 1993 and 1994 respectively. From 1993 to 1995 he worked as a design engineer with GEC-Marconi. He recieved his PhD in 1998 from the Royal Military College of Science, Shrivenham, which is a part of Cranfield University, England. He is currently lecturing at RMCS Shrivenham. His primary research interests include radar cross section modelling and evolutionary algorithms. He is an Associate Member of the Institute of Electrical Engineers and received the prize for best paper at GALESIA 97, Glasgow, UK.