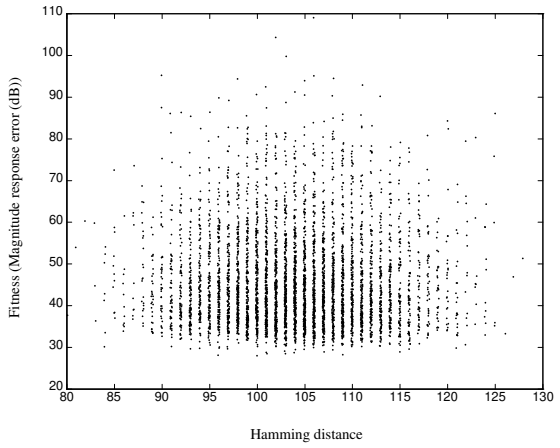

Fig. 6. Scatter plot of fitness against Hamming distance for the random data. This plot has an FDC of 0.0032.

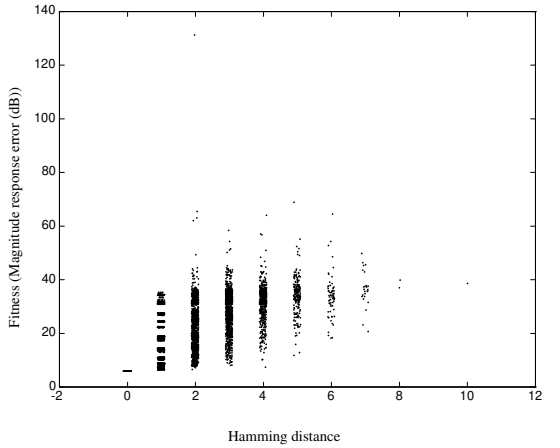

Fig. 7. Scatter plot of fitness against Hamming distance for the 1% perturbed LS data. This plot has an FDC of 0.6466.

ful for this problem. Two measures of GA difficulty, namely the epistasis variance and fitness–distance correlation have also been calculated and have confirmed that the problem is hard, in its present form, for the GA. When far from the optimum, there is little useful structure to direct the GA, as shown by the PS slices and the FDC calculations, so even though the epistasis variance is lower in this region, the GA does not perform well.

Closer to the assumed optimum, the space becomes more structured, and therefore potentially easier to optimise, which is reflected in the increased FDC value. The epistasis variance, however, increases as this region is approached, as the nonlinearity of the representation rises. Consequently the GA is also unable to perform well in this region as crossover and mutation become increasingly disruptive.

Previous work [5][11] has shown that a realcoded GA is suitable for some types of digital filter design. Real coding exhibits far less deceptiveness than a binary alphabet representation, so a restating of the problem with a highercardinality alphabet may prove more successful. It is also possible to reduce the amount of epistasis with some problems, by changing the representation used in the chromosome [12], so further investigation should enable the GA to be a successful optimisation tool in this area.

References

[1] D.E. Goldberg. *Genetic Algorithms in Search, Optimisation and Machine Learning.* Addison Wesley, 1989.

[2] J.H. Holland. *Adaptation in Natural and Artificial Systems.* Ann Arbor, University of Michigan Press, 1975.

[3] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C (2nd Ed.).* Cambridge University Press, 1992.

[4] T. Çiloğlu and Z. Ünver. 'A New Approach to Discrete Coefficient FIR Filter Design by Simulated Annealing'. *Procs. IEEE International Conference on Acoustics, Speech and Signal Processing*, 3:C101–C104, 1993.

[5] S.P. Harris and E.C. Ifeachor. 'Automating IIR Filter Design by Genetic Algorithm'. *Procs. Galesia International Conference*, 1995.

[6] O.V. Patiakin, B.S. Zhang, G.D. Cain, and J.R. Leigh. 'An Adaptive Filter Using Darwinian Algorithm for System Identification and Control'. *Procs. IEEE International Conference on Systems, Man and Cybernetics*, 4:627–631, 1993.

[7] C.J. Rouse and A.J. Carter. 'Exploring Delay/Area TradeOffs of an LDI Filter Using a Natural Based Algorithm'. *Procs. IEEE International Symposium on Circuits and Systems*, 1:A217–A220, 1994.

[8] J.D. Schaffer. 'Multiple Objective Optimisation with Vector Evaluated Genetic Algorithms'. *International Conference on Genetic Algorithms 1985, Proc.*, pages 93–100, 1985.

[9] Y. Davidor. 'Epistasis Variance: A Viewpoint on GA Hardness'. *Foundations of Genetic Algorithms*, pages 23–35, 1990.

[10] T. Jones and S. Forrest. 'Fitness Distance Correlation as a Measure of Problem Difficulty for Genetic Algorithms'. *ICGA 6*, 1995.

[11] E.C. Ifeachor and S.P. Harris. 'A New Approach to Frequency Sampling Filter Design'. *Procs. IEE/IEEE Workshop on Natural Algorithms in Signal Processing*, 1993.

[12] D. Beasley, D.R. Bull, and R.R. Martin. 'Reducing Epistasis in Combinatorial Problems by Expansive Coding'. *ICGA 5*, pages 400–407, 1993.

have an order close to or equal to the length of the string. However, if a problem exhibits zero epistasis, then genes are unaffected by the values of other genes, and a simple bitwise maximisation procedure can be used.

The epistasis variance calculations involve taking a sample of points from the whole search space, and determining the average fitnesses of the samples which have each bit set and clear. These values are then used to calculate an expected fitness for each sample chromosome, from the bits within it. The differences between the actual and expected fitnesses are used to calculate the epistasis variance.

While the calculations described by Davidor give some idea of the nonlinearity of a representation, there are some problems which the GA finds hard, which have a low epistasis, so this measure alone is not a reliable indicator of GAhardness. Jones and Forrest [10] propose another measure, namely the fitnessdistance correlation (FDC), where by calculating the correlation between a string's fitness and the Hamming distance to a global optimum, they claim that a more reliable estimate of GA difficulty can be obtained. For a problem space where the GA is trying to minimise the fitness, then a correlation of 0–1 is expected, with it equalling 1 for a unimodal fitness increase with distance. If the space is highly multimodal, then a correlation close to zero is expected, as there is no clear relationship between fitness and distance. If the FDC measure is substantially less than zero, then the fitness tends to decrease with distance from the global minimum, and therefore the problem will be misleading.

VI. Difficulties Calculated for the GA

A variety of calculations were also performed to examine the difficulty of the problem with respect to both epistasis variance and FDC, close to and remote from an optimum. The fitness function used was simply the maximum magnitude response error in dB, as this had proved the harder of the criteria for the GA to optimise.

The FDC calculations were initially performed with 4000 random samples averaged over ten runs, producing a correlation of almost exactly zero. The data are shown in Figure 6, where it can be seen that there is no clear correlation between fitness and distance. The Hamming distances have been perturbed slightly to allow the distribution to be seen more clearly. The lack of structure in the plot implies that the search space has no structure itself, which the GA could have used to determine the direction of the optimum.

The calculations were repeated for samples taken close around the assumed optimum, by perturbing an average of 5% and then just 1% of the bits. In these regions, the FDC is found to be around 0.32 and 0.65 respectively.

The 5% data showed more structure, while the 1% data, shown in Figure 7, had the highest correlation. This Figure shows that close to the optimum, the change of fitness with distance is more clear, although the height of the densest region, from 1–4 bits difference, shows that even changing a very few bits can lead to a dramatic loss of performance, the most obvious example being the 2bit perturbed filter with a magnitude response error of over 130dB.

	FDC
Random	0.0032
5% perturbed LS	0.3199
1% perturbed LS	0.6466

Table 1. FDC calculations for various regions of the search space.

	Random	10% perturbed LS
σ_ϵ^2	116.75	363.19
σ_v^2	132.25	112.26
σ_A^2	17.73	631.57
$\sigma_v^2 - \sigma_A^2$	114.52	519.31

Table 2. Epistasis variance (σ_ϵ^2) calculations for various regions of the search space.

These results show not only that the search space becomes more ideal as the optimum is approached, but also that it *only* has the characteristics which the GA needs to be able to work effectively close to this same region. The FDC results are tabulated in Table 1.

Epistasis variance (σ_ϵ^2) calculations were performed for a similar set of regions, namely for randomlyselected points, and for the LS solution with 10% of the bits perturbed. The results for 5000 points averaged over five runs, which are tabulated in Table 2, show the epistasis variance for these regions. The fitness variance σ_v^2 is a measure of the spread of the sample fitnesses around the average sample fitness, while the genic variance σ_A^2 is a measure of the spread of the predicted fitnesses around the true fitness sample average. It can be seen that the predicted, genic variance is very low for the random samples and high for the perturbed LS solution samples. In both cases, but especially for the perturbed LS samples, the predicted fitnesses are quite different from their actual values, showing that the representation has a large nonlinearity between the gene values and the corresponding fitnesses.

The high epistasis of the 10% perturbed LS samples ties in with the requirement to have a welldefined impulse response for a desired FIR filter, as any perturbation from this will result in a rapid loss of performance. Another important result from these calculations is that the problem becomes more epistatic as the optimum is approached, which implies that it is also harder for the GA, due to the effects described above.

VII. Discussion and Conclusions

The PS slices presented above show that the problem is multimodal, and that a simple hillclimber is therefore unfeasible as an optimisation method. The very small region occupied by the optimum and the high number of points in the search space ($> 10^{62}$) means that a random search is unlikely to be successful in a designer's lifetime, so an alternative technique is needed.

The GA was presented as a suitable multiobjective optimisation technique, but in practice has proved unsuccess

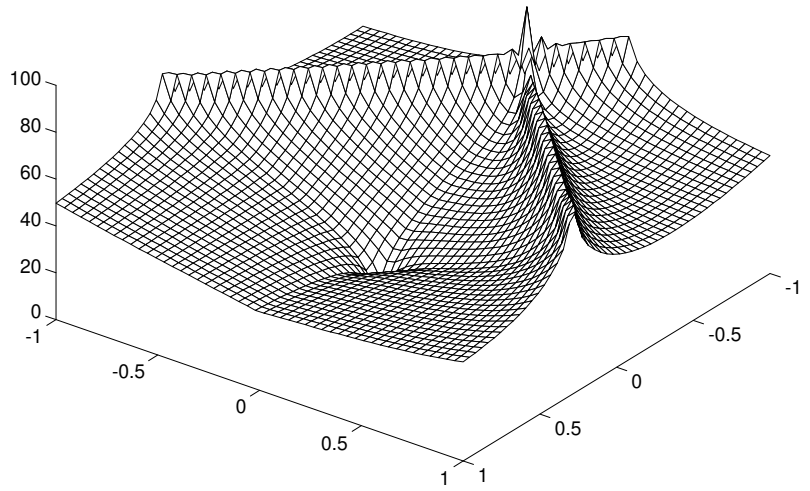

Fig. 3. Slice through the 1,13,LS data set. This slice has a best fitness of 11.3dB.

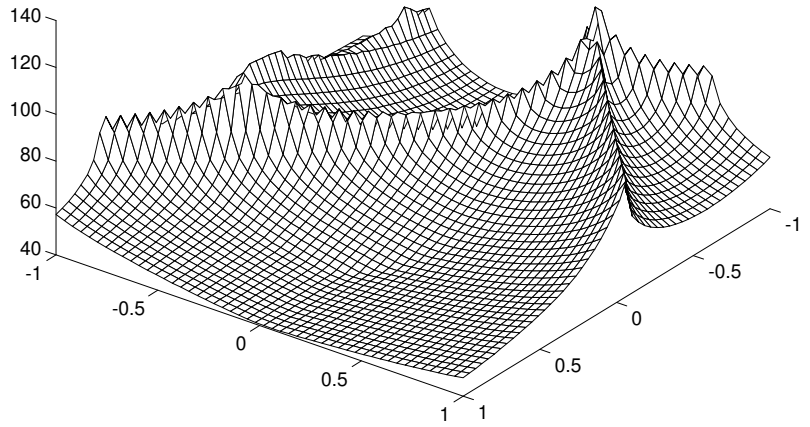

Fig. 4. Slice through the 1,13,P data set. This slice has a best fitness of 40.5dB.

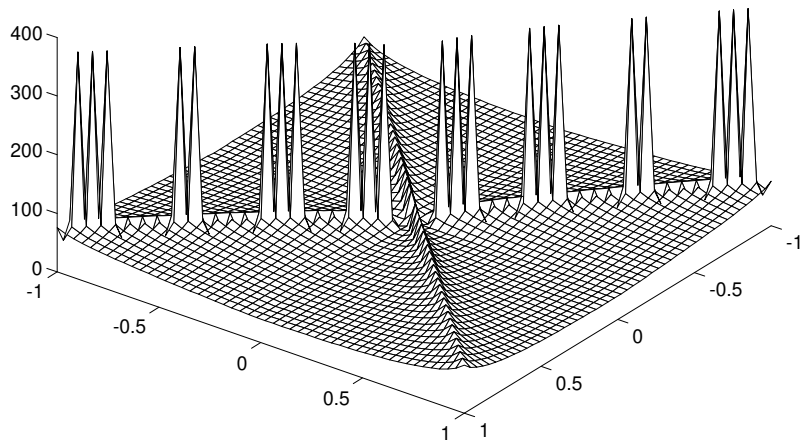

Fig. 5. Slice through the 1,13,R data set. This slice has a best fitness of 32.2dB.

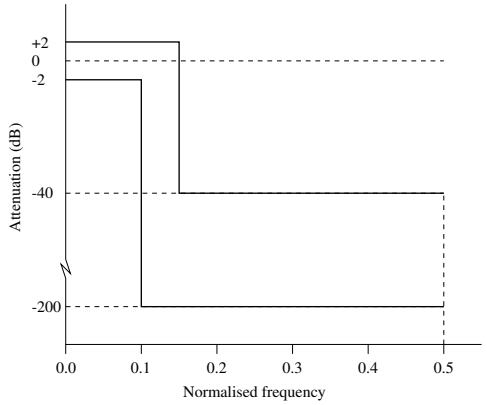

Fig. 2. Loose tolerance filter template used as a test design problem. The linearity of the phase response was examined between normalised frequencies 0–0.05.

nearglobal optimum.

For the 25^{th} order filter, slices were taken by varying the following pairs of coefficients: 1 and 2; 1 and 13; 1 and 24; 12 and 13. The rest of the coefficients were fixed, firstly at the values found by the LS optimisation; secondly at the LS values perturbed by 5% of their value, and finally at ran dom values. For simplicity these data sets will be referred to as LS, P and R, e.g. 1,24,P (Perturbed LS sample points) or 12,13,R (Random points). It should be noted that the changes in the appearance of the PS described below were similar for all the data sets.

Figure 3 shows the 1,13,LS data slice. It shows that the PS consists of smooth higherfitness regions separated by narrow lowfitness ridges. The highfitness region around the optimum is clearly small compared to the total area of the slice, so when this is extended to the full 26dimensional volume, this region will only be a tiny fraction of the whole. The other unperturbed LS slices have similar characterist ics, most with straight ridges as in Figure 3, while 1,24,LS had some curved. In all cases, the optimum region is very small, suggesting that perturbing any coefficient will quickly lead to a large drop in fitness and therefore per formance.

When the 5% perturbed data 1,13,P is used, the surface as shown in Figure 4 becomes more multimodal, with more higherfitness regions of smaller area, separated by similar welldefined, high, narrow ridges. The small highfitness region around the LS optimum has disappeared, leaving a comparatively poor best remaining solution. The position of the best solution available in the space also moves, some times considerably, so seeding the GA with these perturbed solutions will not generally start it off in the region of the true optimum.

The final slices were obtained for a random set of coef ficients. As can be seen in Figure 5, the space is very flat, with very poor filter performance. The narrow, poor fit ness ridges still appear, but the position of the optimum or optima clearly bears no relation to that of the high fit ness region shown in Figure 3, so a GA seeded randomly will have no useful information about where the optimum region is.

V. Measures of GA Difficulty

There are several factors in the structure of the search space and the interactions of the genes in the chromo some which affect how easy the problem is to solve for the GA. *Deceptiveness* can have several causes [1], but has the effect of leading the GA away from the global optimum, e.g. if the global optimum is isolated and surrounded by the worst points in the space, or when combining short, optimal building blocks leads to longer, suboptimal blocks. While deceptiveness is a problem to the GA, some forms, such as the first mentioned above, will also affect all search tech niques which employ some form of hillclimbing or gradient information. It is possible to reduce the deceptiveness by changing to a highercardinality representation, but this is not always possible or desirable if the problem fits naturally into, for example, a binary representation.

For an FIR filter to operate effectively, it has to have a wellformed impulse response (which is the same as the coefficients for this type of filter), and while this might have various characteristics, e.g. symmetry for a linear phase, or have the majority of its amplitude early on for a minimum phase filter, the underlying appearance is similar, with welldefined ripples, whose amplitude rises and falls within a smooth envelope. It is clear, therefore, that for a filter to have a desired response, the value of a particular coefficient is highly dependent on the values of the other coefficients. This in turn implies that it is impossible to predict the performance of a filter from just a few coefficients.

GA schema theory claims that short, highfitness 'build ing blocks' are used to build up good solutions, since chro mosomes containing such a building block should generally have a higher fitness, regardless of the other gene values, and will therefore be selected for retention in the popula tion more often. Since the chromosome to be optimised by the GA is a list of the coefficients, it therefore cannot have any loworder highfitness building blocks, so the GA can not proceed effectively. The strong coefficient coefficient interdependence means that changing the value of one gene in a good solution means that the values of all the other genes must change to maintain the quality of the solution. This causes crossover to be ineffective, as every crossing between two strings which are more than slightly dissim ilar, or which substantially changes the value of even a single coefficient will generally result in offspring with very poor fitnesses. This means that unless a *complete* new solution is found, no great improvement in fitness can be made. Mutation will also be unhelpful unless it is alter ing a lowend bit and therefore only changes the coefficient value slightly.

This interaction between genes, or *epistasis*, is hard to quantify, although Davidor [9] has proposed a measure of the *epistasis variance*. If there are strong interactions, then the linear relationship between gene values and fit ness breaks down, such as with a delta function where the fitness is zero unless all bits are set. This implies that the only high fitness schema in highly epistatic problems

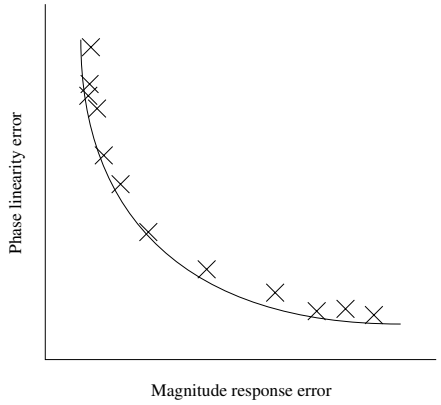

Fig. 1. Example of a possible paretooptimal set for a nonlinear filter design problem.

Another approach, the vectorevaluated GA (VEGA) adopted by Shaffer [8], is to split the population into several subpopulations and evaluate the fitness of each of these by a different criterion. The populations are then recombined before crossover and mutation occur, in order to exchange information between the subpopulations. For this approach to be effective, the individual fitnesses should be scaled into roughly the same range, which it may be hard to achieve for unconstrained problems or those where the parameter constraints are not wellunderstood. VEGA keeps a record of those solutions which 'dominate' the others, i.e. they have a better performance for at least one criterion, and perform at least as well with respect to the rest. This 'nondominated' set (NDS) contains solutions who cannot be said to be 'better' than each other, as an improvement in one measure always causes a decrease in performance with respect to at least one other. If a scatter plot of the overall best solutions in the search space is drawn, then a wavefront of nondominated solutions can be seen, beyond which there are no solutions—this 'pareto optimal' set (POS) cannot be improved on. An example POS for a filter design problem is shown in Figure 1. If this set could be found for a given problem, then the designer would have a whole range of 'best' solutions with differing performance tradeoffs, and could then choose the one which best fitted the problem in hand.

The GA can easily be developed to produce points which approach (or are in) the paretooptimal set [1], by encouraging the population to spread out along the wavefront of nondominated solutions in the current population. These can be saved at each generation, and the overall best NDS found. This approach can be extended to multidimensions, so that a number of design criteria can be optimised simultaneously without user intervention.

III. The Design Problem

In order to utilise the multiobjective optimisation ability of the GA to combine steps in the design of digital filters, a GA was developed to find the POS for an FIR filter with nonlinear phase, where the criteria to be minimised are the maximum error between the actual response and the desired response template, and the χ^2 error from linearity of the phase response in a specified region of the passband. The chromosome was a binary string containing a concatenated list of the coefficient values, which therefore intrinsically accounts for finite wordlength effects.

Initial runs were performed with a random initialisation, and proved unable to find a solution which fitted magnitude response templates to any great accuracy, even to within 10–20dB, although the optimisation of the phaselinearity was generally more successful. Since reducing the minimax error is a harder problem, the GA was changed to optimise the RMS error in the magnitude response. This proved more successful, although the maximum error of the filters found was greater than before.

To 'kickstart' the GA in the region of a known good design, a Remez exchange solution for the desired filter was used as a seed for one member of the population, and the rest filled with perturbed copies of it. This approach, however, caused premature convergence in the population, as the Remez exchange solution is either very close to or at the nearest optimum. To combat this, the seeds were dispersed away from the Remez exchange design by mutating an increasing proportion of the bits within them. This allowed the GA to search more effectively, but it was never able to find a good solution even as good as the original quantised Remez exchange design, which it was expected to improve on as the linear phase constraint had been relaxed.

Since the GA had not performed well, a very loose tolerance filter template as shown in Figure 2 was used as a test problem. The order (as determined by Matlab) was set to 25, and the wordlength to eight. As the order was determined for a linear phase filter, and the filter being designed was not linear phase over the whole response, it was anticipated that the GA should be able to find an acceptable solution to this problem, as more degrees of freedom were available for the optimisation of the magnitude response. Unfortunately, this was again unsuccessful, the GA only being able to improve a little on the initial best fitnesses.

At this point, investigations were undertaken to determine why the GA was not performing well, and to analyse the difficulty of the problem. The analyses, which were performed for the filter design template in Figure 2 will now be detailed.

IV. Analysis of the Parameter Space

In order to gain a better understanding of the nature of the space the GA is searching, 'slices' were taken through the Parameter Space (PS) in a number of locations, for the filter specification shown in Figure 2. The fitness function used was the maximum magnitude response error, so a lower fitness value means a better filter. To draw each slice, all but two of the filter coefficients were fixed, and a surface plotted from the fitnesses found by varying the others. To find a highfitness region, in the absence of a solution found by the GA, coefficients calculated by the Remez exchange method were used to seed a bitflipping local search (LS) algorithm, and the solution this found was taken to be a

Nonlinear FIR Filter Design by Genetic Algorithm

Stephen P. Harris
SECEE
University of Plymouth
Drake Circus, PLYMOUTH
Devon, UK, PL4 8AA
steve@cis.plym.ac.uk

Emmanuel C. Ifeachor
SECEE
University of Plymouth
Drake Circus, PLYMOUTH
Devon, UK, PL4 8AA

Abstract— **In this paper, an approach to digital filter design which reduces the number of design steps is proposed, whereby the multiobjective optimisation ability of the Genetic Algorithm is utilised to simplify the design procedure. The technique is designed to produce a range of solutions with different performance tradeoffs, from which the designer can select the most appropriate. An analysis of the parameter space for an FIR filter design problem is presented, with a number of investigations into its difficulty. Reasons for the failure of the GA to produce satisfactory results are discussed.**

I. Introduction

The design of digital filters, as with most engineering tasks, is a multistep, iterative process. The key stages are filter specification, coefficient calculation, structure realisation, finite wordlength analysis, and implementation. Each stage involves optimisation, but current practice is to perform this separately for each stage in an iterative fashion until an acceptable solution is found. However, since the effects of each step are interrelated, optimisation at only one step leads to a design which, while optimal for that design step, will generally be suboptimal for the others. A desirable goal is to perform the optimisation over all design steps simultaneously, in order to seek out the filter design with the best overall tradeoff across all of the design criteria.

Finite Impulse Response (FIR) filters are particularly useful to designers who need highly linear phase systems, such as in biomedicine, where it is of paramount importance to avoid distorting the signal during filtering. FIR filters can be made to have exactly linear phase by forcing them to have a symmetric impulse response (such as those designed by the Frequency Sampling method), so are often preferable to Infinite Impulse Response (IIR) filters where linear phase can only be approximated. They are also simpler to design, for example frequency sampling method filters can be designed with a simple hillclimber. A disadvantage of nonrecursive FIR filters is that they require many more coefficients to achieve the same performance as an equivalent IIR, but if a nonlinear filter can be used instead, the extra degrees of freedom permit a saving of up to half the number of filter coefficients while retaining the same magnitude response.

To obtain a more efficient linear phase FIR filter, the linear phase region can be confined to a portion of the passband, and the phase left unrestricted in the stopband where it is of less importance. This will allow a filter to be designed with fewer coefficients than for a fully linear phase system, but does introduce another design criterion into the filter design process. To design this type of filter requires both the magnitude and phase responses to be optimised. This is a multicriterion optimisation problem, which means that standard methods such as hillclimbing are no longer suitable, and an alternative must be found.

In recent years, natural algorithms such as the Genetic Algorithm (GA) [1][2] and Simulated Annealing (SA) [3][4] have become popular optimisation tools for performing searches of hilly, multidimensional surfaces where traditional methods such as hillclimbing cannot perform well. Both of these techniques avoid the need for specialised expressions, and no detailed analysis of the problem need be performed. These features can be utilised in digital filter design to perform various optimisations [4][5][6][7].

In this paper, an approach to combining filter design steps by using a multiobjective GA to optimise the magnitude and phase response of nonlinear phase, quantised coefficient FIR filters is presented. A number of analyses performed on the problem in the light of poor GA performance are also detailed.

II. Multiobjective optimisation by GA

The GA is an optimisation technique which is inspired by the concepts of Darwinian evolution through reproduction and natural selection. It stores a number of representations of solutions to a problem in a 'population', and exchanges information between pairs of them to produce 'offspring', which are then retained according to how well they fulfil the design criteria of the problem under investigation. This performance measure is known as the 'fitness'. This process is repeated until an acceptable solution has been found or a preset number of iterations or 'generations' have been performed.

The simplest way to perform multiobjective (or multicriterion) optimisation is to combine the individual fitness measures into a single value by addition or multiplication, although this technique is not straightforward for complex problems. In order to allow the importance of each criterion to be controlled, some form of weighting scheme needs to be introduced, and many runs performed until a satisfactory weighting (and suitable solution) are found. While it may be possible to employ a 'metaoptimisation' technique to optimise the weighting, this still requires many times the number of function evaluations of a single run.