# Behavioural Performance and Variation Modelling for Hierarchical-based Analogue IC Design

Sawal Ali, Reuben Wilcock, Peter Wilson
Electronics System Design Group, School of Electronics and Computer Science
University of Southampton, UK
{shma05r, rw3, prw}@ecs.soton.ac.uk

## ABSTRACT

A new approach in hierarchical optimisation is presented which is capable of optimising both the performance and yield of an analogue design. Performance and yield trade offs are analysed using a combination of multi-objective evolutionary algorithms and Monte Carlo simulations. A behavioural model that combines the performance and variation for a given circuit topology is developed which can be used to optimise the system level structure. The approach enables top-down system optimisation, not only for performance but also for yield. The model has been developed in Verilog-A and tested extensively with practical designs using the Spectre simulator. A benchmark OTA circuit is used to demonstrate the behavioural model development and a 7th order video filter has been designed to demonstrate hierarchical optimisation at the system level. The results have been verified with transistor level simulations and suggest that an accurate performance and yield prediction can be achieved with the proposed algorithm.

## INTRODUCTION

Advances in silicon technology over the last decade have led to increased integration of analogue and digital functional blocks onto the same chip. In such a mixed signal environment, the analogue circuits must use the same transistors as their digital neighbours. The increasing complexity and accuracy of device models has led to wide acceptance of simulation and optimisation based design techniques for the design of analogue blocks rather than hand calculations [1~4]. With reducing transistor sizes, the impact of process variations on analogue design has become very prominent and can lead to circuit performance and yield falling below specification. Due to the high correlation of circuit yield to profit, yield maximisation must be considered early in the design process, and this has led to the concept of design for yield (DFY) [5].

In optimisation-based design techniques, the performance of the circuit must be evaluated for a large number of different circuit variables, a process is known as design space exploration. Running the entire performance evaluation at transistor level is computationally intensive especially for large and complex circuits. Due to this limitation, hierarchical design is commonly used to break down a large system into its constituent building blocks. A typical hierarchical design is shown in figure 1. Not only does this approach simplify the design task but it also speeds up the design flow by encouraging design reuse. Hierarchical based optimisation is one method used to reduce simulation time and involves the use of behavioural models prior to transistor level simulation [6~9]. However, all previous works in hierarchical optimisation approaches do not consider yield in the design process. This paper proposed a novel hierarchical optimisation methodology that combines behavioural performance and statistical variation in the design algorithm.

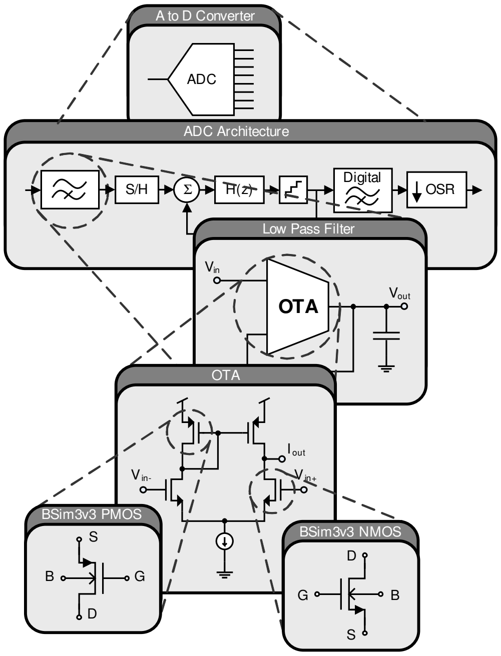


**Figure 1. A typical system design hierarchy**

The paper is organised as follows: section 2 provides necessary background; the hierarchical optimisation architecture is detailed in section 3 and experimental examples in section 4. Concluding remarks are given in section 5.

## BACKGROUND

Simulation-based optimisation techniques are widely used for analogue circuit design and several synthesis tools have been developed that use spice-like simulators for their evaluation engine. However, these approaches are processor intensive which limits their use to smaller building blocks [10]. To overcome this problem, hierarchical based design has been proposed to divide the large system into sub blocks that can be optimised separately [10].

## 1.1Multi Objective Optimisation

The optimisation formulation for more than one objective function is called multi-objective optimisation (MOO) which can be generally stated as:

$$\begin{aligned} &Minimise\,/\,Maximise\ f_m(x),\ m=١,٢...M \\ &Subject\ to\ g_j(x)\geq ٠,\ j=١,٢...J \end{aligned} \tag{1}$$

Where $f_m(x)$ is the set of $M$ performance functions and $g_j(x)$ is the set of $J$ constraints and the outcome is a set of optimal solutions [11]. Figure 2 shows the relationship between the parameter space and objective space, where each point in the parameter space is a solution that corresponds to a point in the objective space. The black curve shown on the objective space is called the *Pareto front* and all solution points lying on this curve are called Pareto-optimal solutions. The algorithm used in this work for the MOO is called Non-dominated Sorting Genetic Algorithm -II (NSGA-II). This evolutionary algorithm employs an elite preserving strategy which makes sure that good design solutions found early in the optimisation will be carried to the next generation. The following shows an outline flow of an NSGA-II algorithm:

```
NSGA Algorithm

- Generate initial random population, size N.
- Create offspring population
- Combine parent and offspring population to form $R_t$ ($R_t=P_t \cup Q_t$)
- Perform non-dominated sorting and identify fronts, $F_i$ (i=1,2...etc)
- Set new population, $P_{t+1} = 0$, and fill $P_{t+1}$ with $F_i$ ,($P_{t+1} \cup F_i$) as long as $|P_{t+1}|+|F_i|<N$.
- Perform crowding sort and place most widely spread solution in $P_{t+1}$
- Create offspring populaiton $Q_{t+1}$ from $P_{t+1}$ and repeat until last number of generation.
```

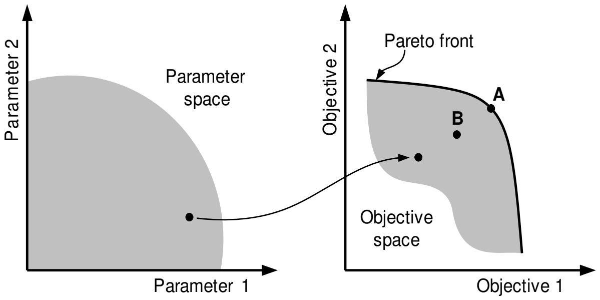
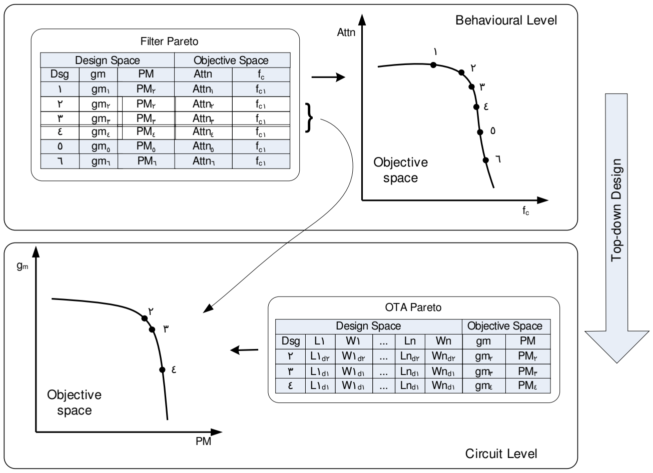


**Figure 2. Parameter space and objective space.**

## 1.2Hierarchical Based Design

A hierarchical methodology consists of a top-down design and bottom-up verification process [12]. Important aspects of both processes are circuit decomposition and specification propagation. Once the system architecture has been divided into sub-blocks, automatic optimisation algorithms can be applied to solve for the circuit sizing. The optimisation for hierarchical based design can be divided into two steps: In step one, the behavioural level blocks are optimised using a behavioural simulation and the design parameters that meet the system level specification are determined. In the second step, the design parameters from the previous optimisation are taken as the specifications for the circuit level optimisation which propagates the system level specification to the bottom level. The relationship between transistor level and system level optimisation in hierarchical design is shown in figure 3. The behavioural-level Pareto front determines all the solution points that meet the system level filter specifications. The design space of this Pareto front is then taken as the design objective for the sub-block circuit level. At the circuit level, the Pareto front is then used to determine the design parameters that best meet the design objective, resulting in the transistor dimensions.



Filter Pareto

| Design Space | | | Objective Space | |
|---|---|---|---|---|
| Dsg | gm | PM | Attn | $f_c$ |
| ١ | $gm_١$ | $PM_٢$ | $Attn_١$ | $f_{c١}$ |
| ٢ | $gm_٢$ | $PM_٢$ | $Attn_٢$ | $f_{c١}$ |
| ٣ | $gm_٣$ | $PM_٣$ | $Attn_٣$ | $f_{c١}$ |
| ٤ | $gm_٤$ | $PM_٤$ | $Attn_٤$ | $f_{c١}$ |
| ٥ | $gm_٥$ | $PM_٥$ | $Attn_٥$ | $f_{c١}$ |
| ٦ | $gm_٦$ | $PM_٦$ | $Attn_٦$ | $f_{c١}$ |

OTA Pareto

| Design Space | | | | | | Objective Space | |
|---|---|---|---|---|---|---|---|
| Dsg | L١ | W١ | ... | Ln | Wn | gm | PM |
| ٢ | $L١_{d٢}$ | $W١_{d٢}$ | ... | $Ln_{d٢}$ | $Wn_{d٢}$ | $gm_٢$ | $PM_٢$ |
| ٣ | $L١_{d١}$ | $W١_{d١}$ | ... | $Ln_{d١}$ | $Wn_{d١}$ | $gm_٣$ | $PM_٣$ |
| ٤ | $L١_{d١}$ | $W١_{d١}$ | ... | $Ln_{d١}$ | $Wn_{d١}$ | $gm_٤$ | $PM_٤$ |

**Figure 3. Hierarchical Optimisation**

Macromodelling is a useful technique that involves developing models from simulation data points. The performance model that relates the performance and circuit parameters obtained from macromodelling can be used in hierarchical design. The primary advantage of developing hierarchical macromodels is that the macromodels need only be built once. They can be used during design time and the performance of a block at any time can be determined without having to travel to the very bottom of the hierarchy.

# PROPOSED ALGORITHM

The key steps in the proposed algorithm for the performance and variation model development are shown in Figure 4. These steps are now discussed in more detail.

## 1.3Netlist and Objective Function Generation

The starting point for the proposed algorithm is a circuit topology, process models and a set of performance functions. The first step involves generating a transistor level netlist for the chosen circuit topology. From this netlist a set of designable parameters are derived which will be used to change the circuit's performance. Examples of designable parameters include a transistor's length and width. Each parameter will have constraints imposed by the designer that define the decision space for the optimisation. The performance functions of the circuit are defined as the objective functions. Testbench netlists are defined to simulate the performance for a certain set of parameters.

## 1.4Multi-Objective Optimisation

In this stage, the design space is explored and the objective functions are improved iteratively. The optimisation implementation is based on an evolutionary algorithm known as Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [11]. The genetic algorithm procedure involves generating a number of individuals and optimising these over a number of generations. The individuals are encapsulated in a set of parameters defined as the GA string. During the optimisation, the algorithm determines the quality of the individuals through the fitness score of each individual. The fitness score is measured from the performance evaluation.

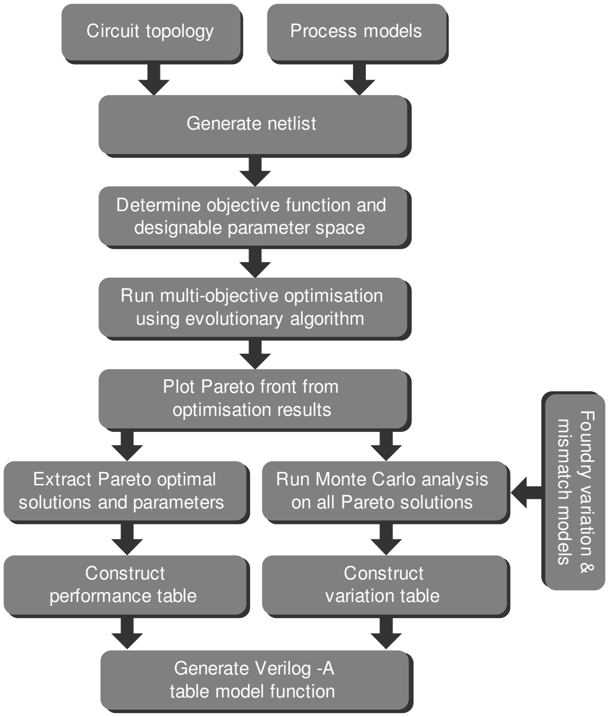

Circuit topology
Process models
Generate netlist
Determine objective function and designable parameter space
Run multi-objective optimisation using evolutionary algorithm
Plot Pareto front from optimisation results
Extract Pareto optimal solutions and parameters
Run Monte Carlo analysis on all Pareto solutions
Foundry variation & mismatch models
Construct performance table
Construct variation table
Generate Verilog -A table model function

**Figure 4. Novel yield targeted algorithm.**

## 1.5Performance and Variation Modelling

The outcome of the multi objective optimisation is a Pareto-front (a set of optimal solutions). All the solution points on the Pareto-front and their respective design parameters can be extracted and a model can be created that represents these data points. Interpolation is one of the techniques that can be used to model these data points. In this technique, the number of fitting parameters in the extraction process matches the number of samples in the data points meaning that all the data points are used for the model. These data points can be stored in a look-up table and interpolation can be applied to the table to find intermediate design points. The Verilog-A behavioural language is used for this process due to its support for lookup-tables and interpolation. A performance model of a circuit design is a model that relates the design performance with design parameters. Having obtained the Pareto-points, all the optimal solutions and their parameters are stored in a data file which defines the optimal performance model for the design.

It is important to consider process variation as early as possible in the design flow. Such variations can cause a circuit's performance to vary from their nominal point, reducing overall yield. This is a very important step in hierarchical-based design for yield prediction. The specifications in hierarchical design are given for the system level, however the yield of the system is influenced by the variations in the sub-block circuits. Therefore the performance spread of the sub-block circuit needs to be predicted and yield of the whole system optimised. Monte Carlo (MC) analysis is the best candidate for this purpose [13]. Therefore, during this step, a MC analysis is run for each of the parameter solution sets that lies on the Pareto-front. From this simulation, a set of performance spreads is obtained. The performance spread information is stored together with the performance model in a datafile.

## 1.6Behavioural Model Development

This step in the proposed architecture involves developing a behavioural model description for the circuit block so that it can be used in system level optimisation. The performance and variation model developed in the previous stage will be defined in behavioural language and will be added in the behavioural description of the circuit block. The performance and variation model are defined as a look-up table using the table model function in Verilog-A. This function allows the module to approximate the behaviour of a system by interpolating between the performance and variation data points extracted from the Pareto-front. The syntax of the table model function is shown below:

```
$table_model(f1,f2, “datafile.tbl”, “control_string”);
```

Where f1 and f2 are the performance functions, ‘datafile.tbl’ is the text file that contains the performance functions and design parameters, and ‘control string’ determines the interpolation and extrapolation method. In this algorithm, a cubic spline method is used for the interpolation and no extrapolation method is used, in order to avoid approximation of the data beyond the sampled data points. The algorithm creates a table model function for both performance and variation functions.

# 2.EXPERIMENTAL EXAMPLES

This section presents a complete design example using a single stage operational transconductance amplifier (OTA) as the target circuit for the behavioural model development. OTAs are fundamental building blocks, often employed in analogue circuit design applications such as filters. All the following simulations were performed using the industry standard Cadence Spectre simulator with foundry level BSim3v3 transistor models from a standard 0.12um ST process.

## 2.1Design Setup

The initial chosen circuit topology is the single stage OTA shown in figure 5. The first step is to determine the designable parameters for the topology. In this example, these are the transistor lengths and widths which make up a total of 4 designable parameters. Three objective functions have been chosen for this example: transconductance (gm), output resistance (ro) and phase margin (pm). A testbench netlist was created to evaluate the performance functions.

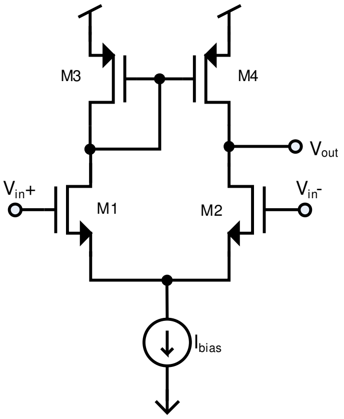


**Figure 5. Single Stage OTA topology.**

## 2.2Multi-Objective Optimisation

The designable parameters must be constrained within a reasonable range which defines the design space of the optimisation. In this case all transistor lengths and widths were specified to be between 0.12µm-4µm and 0.12µ-120µm. Once the parameters have been determined, a GA string can be constructed and is shown in figure 6. A Genetic Algorithm then generates the design parameters according to this string and it is these parameters that are used in the Spice netlist for the simulations. A total of 50 generations each with a population size of 400 were used in this case, giving 20,000 total samples for the optimisation.

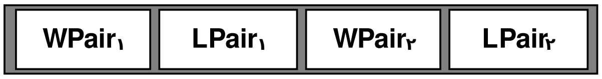


**Figure 6. GA string for the design example.**

The testbench netlist is used to evaluate the performance for each design parameter set (defined by GA) and the result of the simulations determines the fitness score of the individuals. A non-dominated sorting and crowding distance sorting are applied to the solution for each generation in order to find the final diverse set of Pareto-fronts. The result of the optimisation is a full set of designable parameters and their performance functions.

## 2.3Performance and Variation Modelling

The outcome of the MOO for the OTA is a Pareto-front consisting of 211 solution points which defines the performance model of the circuit. To develop the variation model of the Pareto-front, every optimal solution undergoes a Monte Carlo simulation using process variation and mismatch models. 200 samples were chosen for the MC simulation and from these the variation for each performance is calculated. This completes the variation model and the results are stored in a data file. At this point, a combined performance and variation model for the OTA is developed. A look-up table is defined for the table model function in the Verilog-A model given in listing 1. Table 1 shows a selection of samples points from the table and their performance and variation values.

**Listing 1. Performance and Variation Model**

```
analogue begin
 gm_delta = $table_model (gain, "gm_delta.tbl", "3E");
 ro_delta = $table_model (ro, "pm_delta.tbl", "3E");
 pm_delta = $table_model (pm, "pm_delta.tbl", "3E");
 gm_prop = ((gm_delta/100)*gm)+gm;
 ro_prop = ((ro_delta/100)*ro)+ro;
 pm_prop = ((pm_delta/100)*pm)+pm;
 p1 = $table_model (gm_prop,ro_prop,pm_prop
 "p1_data.tbl","3E,3E,3E");
 p2 = $table_model (gm_prop,ro_prop,pm_prop
 "p2_data.tbl","3E,3E,3E");
 p3 = $table_model (gm_prop,ro_prop,pm_prop
 "p3_data.tbl","3E,3E,3E");
 p4 = $table_model (gm_prop,ro_prop,pm_prop
 "p4_data.tbl","3E,3E,3E");
 fptr=$fopen("params.dat");
 $fwrite(fptr, "\n Generated Design Parameters\n ");
 $fwrite(fptr, "%e %e %e %e", p1,p2,p3,p4);
 $fclose(fptr);
 $display ("params: = %e %e %e %e", p1, p2, p3, p4);
End
```

**Table 1. Performance and Variation Samples**

| Design: | gm : | Δgm: | ro : | Δro: | pm: | Δpm : |
|---|---|---|---|---|---|---|
| 2 | 109µ | 0.75% | 382k | 0.75% | 87.9 | 1.74% |
| 3 | 109µ | 0.75% | 384k | 0.75% | 87.8 | 1.73% |
| 19 | 110µ | 0.74% | 371k | 0.74% | 88.0 | 1.73% |
| 34 | 111µ | 0.75% | 497k | 0.74% | 85.3 | 1.71% |
| 35 | 111µ | 0.73% | 375k | 0.75% | 87.9 | 1.73% |
| 61 | 112µ | 0.73% | 458k | 0.74% | 86.1 | 1.71% |
| 209 | 120µ | 0.70% | 486k | 0.74% | 82.7 | 1.70% |
| 211 | 120µ | 0.70% | 743k | 0.72% | 74.9 | 1.69% |

## 2.4Behavioural Description

In hierarchical optimisation, behavioural modelling is used for the system level optimisation and offers the basis for a fast selection process. In this example, a behavioural model for an OTA is developed based on ac small signal analysis. For this purpose, a high frequency model is created, shown in Figure 7. All the transistor parasitic capacitances, including the gate source, and drain overlap capacitances are combined into a single elements $C_{in}$ and $C_{out}$ to provide a greater accuracy in the modelling. The resulting behavioural description, written in Verilog-A is given in listing 2.

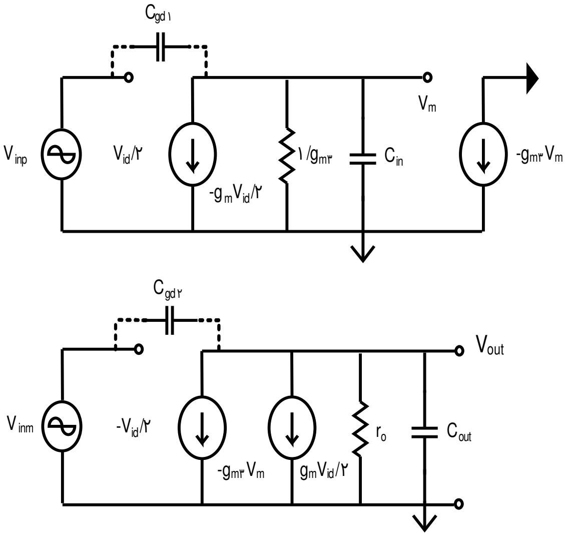


**Figure 7. OTA Small Signal Model**

**Listing 2. OTA Behavioural Model**

```
Module ota(inp, inm, out)
   ….
   parameter real gm = 60e-6;
   parameter real ro = 1e+6;
   electrical inp, inm, out, vm;
   real vin;

analog begin
   // high frequency model
   vin = V(inp,inm);
   I(vm) <+ -gm*(vin/2);  // gm transistor M1
   I(vm) <+ cin*ddt(V(vm)); // cin is the total input stage capacitance
   I(vm) <+ cgd1*ddt(vin/2);  // miller effect of cgd1
  …
   I(out) <+ -gm3*V(vm);
   I(out) <+ -gm*(vin/2);
   …..
   V(out) <+ I(out)*ro;
   …..
end
endmodule
```

## 2.5Hierarchical Optimisation

The behavioural model developed for the OTA is used in the system level optimisation of a 7th-order Elliptic video filter. The designable parameters for the filter are the OTA transconductance (gm) and capacitor values (C1~C10). The specification for this example is based on typical video filter requirements which are shown in Figure 8 [14]. These specifications define the objective space for the optimisation. The filter topology is shown in figure 9.

Multi objective optimisation using NSGA-II algorithm is then performed on the filter design to locate an optimum solution point. A total of 200 individuals and 50 generations were used for the optimisation process. Table 2 shows a number of sample design points, and their performance parameters.

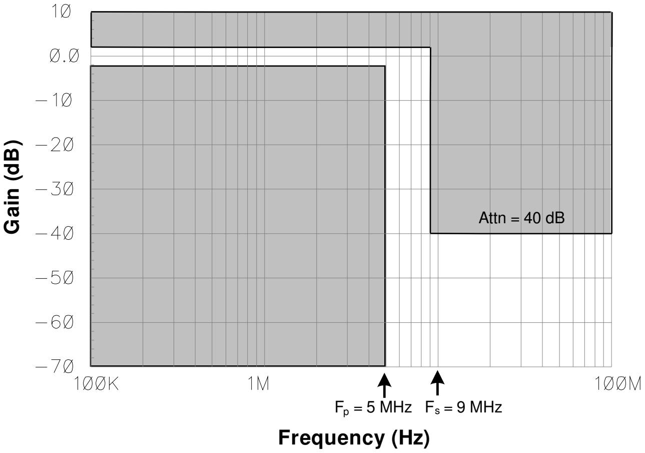


**Figure 8. Elliptic filter specification**

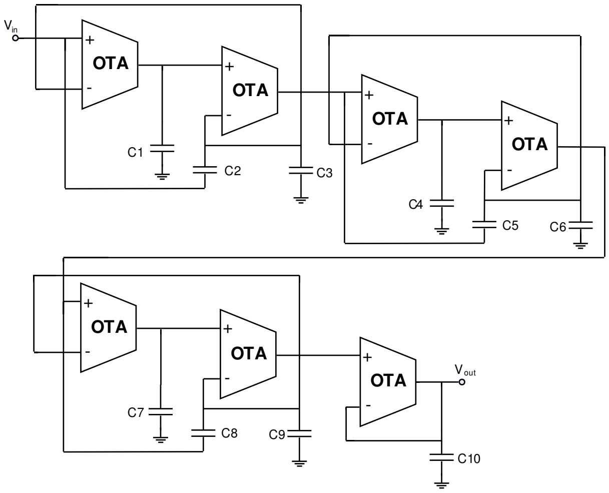


**Figure 9. 7th Order Elliptic Filter Schematic**

**Table 2. Pareto-front Samples for Filter**

| Design: | gm (µ) : | Attn (dB): | Fp(MHz): | Fs(MHz): |
|---|---|---|---|---|
| 11 | 122.3 | 40.3 | 6.1 | 8.3 |
| 22 | 131.6 | 47.4 | 5.4 | 7.5 |
| 15 | 108.9 | 45.9 | 5.3 | 7.3 |
| 70 | 113.8 | 55.1 | 5.7 | 8.9 |
| 61 | 130.4 | 61.7 | 5.7 | 8.9 |

The results in table 2 show all the design solutions that meet the filter specifications. The design parameters (i.e. gm) for these solutions will be taken as specification for the OTA. Based on the performance and variation model of the OTA, the only feasible solutions for the filter are design points 15 and 70. The other design points require a higher transconductance value which is not feasible for the OTA topology.

The variation model already generated for the OTA is used to interpolate the transconductance variation and from this interpolation, the maximum and minimum transconductance can be determined. From table 1, it can be seen that the variation for design number 15 (gm=108.9µ) and 70 (gm=113.8µ) is 0.75% and 0.73% respectively. The maximum and minimum range for both of the gm obtained from the variation value will be used in the filter simulation in order to verify the yield of the filter. From the analysis, design point 15 gives 100% yield compared to design point 70 and therefore point 15 will be used for the OTA parameters in the final filter design. The result of the hierarchical optimisation is a filter that has been optimised to meet high level specifications taking process variations into consideration. To verify the predicted yield given by the proposed approach, a Monte Carlo analysis with 100 samples was run on the final design. This analysis confirmed a yield of 100% as shown in figure 10.

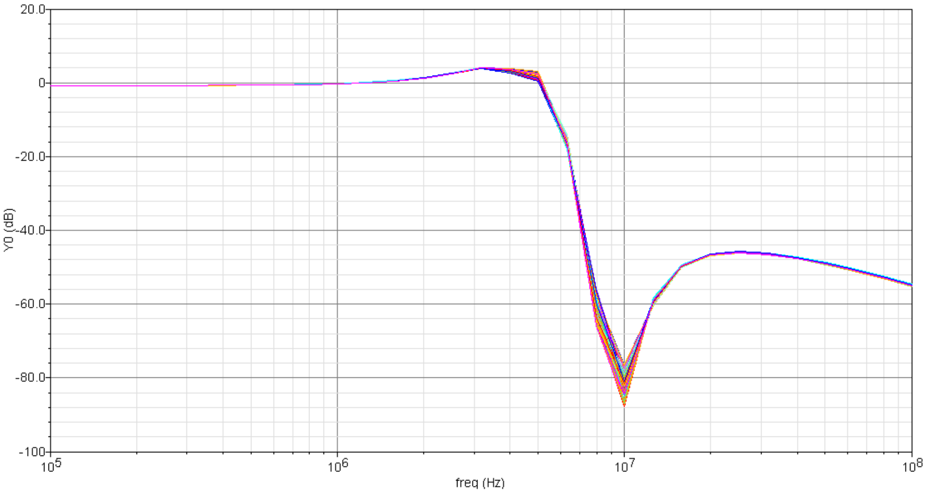


**Figure 10. Monte Carlo Plot of Filter response**

# 3.CONCLUSIONS

This paper has presented a new approach for hierarchical-based design that combines performance and yield optimisation for an analogue system. Multi-objective optimisation based on an evolutionary algorithm is used to explore tradeoffs between performance and yield, leading to a set of Pareto optimal solutions for the design. Monte Carlo variation analysis is performed on all the Pareto optimal solutions, and a table is constructed for both the performance and variation analysis. A behavioural model developed in Verilog-A is used together with this table to determine the parameters required to achieve the highest yield within a given specification. The behavioural model is used for a system level simulation and the approach demonstrates a successful top-down optimisation. These benefits are enjoyed without a corresponding drop in accuracy. A benchmark OTA topology and video filter design were used to demonstrate the proposed algorithm and the behaviour has been verified with transistor level simulations.

# ACKNOLOWLEDGMENTS

The authors wish to acknowledge the support of Cadence Design Systems (UK), particularly Paul Chapman and Ian Clifford, and to gratefully thank them for their support. Without this support, this work would have been much more difficult to undertake. The authors also wish to note the support of the Engineering and Physical Sciences Research Council (EPSRC) for funding of key staff (GR/S95770, EP/E035965).

# REFERENCES

[1] B.De Smedt and G.Gielen, “Watson: design space boundary exploration and model generation for analogue and rfic design,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transaction on*, vol.22, no.2, pp. 213-224, Feb. 2003.

[2] Stehr G., Graeb H., and Antreich K., “Performance trade-off analysis of analog circuits by normal-boundary intersection.,” *in Proc. Of Design Automation Conferenc 2003,* pp. 958-963.

[3] E.S. Ochotta, R.Rutenbar, and L.R. Carley, “Synthesis of high-performance analogue circuits in ASTRX/OBLX,” *IEEE Trans. Computer -Aided Design,* vol. 15, pp.273-294, Mar. 1996

[4] M. Krasnicki, R.Phelps, J.R.Hellums, M. McClung, R.A Rutenbar and L. Richard Carley, “ASF: a practical simulation-based methodology for the synthesis of custom analogue circuits,” in Proc. ICCAD 2001, pp 350-357.

[5] B. D. Smedt, G. Gielen, “HOLMES: Capturing the yield-optimized design space boundaries of analogue and RF Integrated Circuits.” In *Proc. Of the Design,Automation and Test* in Europe Conference and Exhibition, 2003.

[6] Tiwary S.K., R.R.M.T., “Pareto Optimal Modeling for Efficient PLL Optimization,” *in Technical Proceeding, 2004 NSTI Nanotechnology conference and Trade Show, 2004.*

[7] J. Zou, D. Mueller, H. Graeb, U. Schlichtmann, E. Hennig, and R. Sommer, “Fast Automatic Sizing of a charge pump phase-locked loop based on behavioural models,” *in Proc. IEEE International Behavioural Modeling and Simulation Workshop BMAS 2005*, 22-23 Sept. 2005, pp. 100-105.

[8] J. Zou, D. Mueller, H. Graeb, U. Schlichtmann, “A cppll hierarchical optimization methodology considering jitter, power and locking time,” *in Proc. 43rd ACM/IEEE Design Automation Conference*, 24-28 July 2006, pp. 19-24.

[9] J. Zou, D. Mueller, H. Graeb, U. Schlichtmann, E. Hennig and R. Sommer, “Pareto-front computation and automatic sizing of cpplls,” *in Proc. 8th International Symposium on Quality Electronic Design ISQED ’07*, 26-28 March 2007, pp. 481-486.

[10] T.Eeckelaert, T. McConaghy and G. Gielen, “Efficient multiobjective synthesis of analog circuits using hierarchical pareto-optimal performance hypersurfaces,” *in Proc. Design, Automation and Test in Europe*, 2005, pp. 1070-1075.

[11] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons Ltd, 2001.

[12] R.A. Rutenbar, and G. Gielen, “Computer-aided design of analog and mixed-signal integrated circuits,” *Proceedings of the IEEE*, vol. 88, pp. 1825-1852, 2000.

[13] R. Spence and R.S. Soin, “Tolerance Design of Electronic, Addison Wesley”, 1988.

[14] I. Bezzam, C. Vinn, and Rao Rangaiya, “A fully-integrated continous-time programmable CCIR 601 video filter,” *in International Conference of Solid-State Circuits 1995, digest of Technical Papers, 42nd ISSCC*, 15-17 Feb 1995, pp. 296-297, 383.