Behavioural Modelling of Operational Amplifier Faults using analogue Hardware Description Languages

Peter R. Wilson[1], Yavuz Kiliç † (Member IEEE), J. Neil Ross, Mark Zwolinski (Senior Member IEEE), Andrew D. Brown

University of Southampton,
Department of Electronics and Computer Science,
Southampton, United Kingdom,
prw99r@ecs.soton.ac.uk

†Philips Semiconductors,
Southampton,
United Kingdom
Yavuz.Kilic@philips.com

Abstract

The use of behavioural modelling for operational amplifiers has been well known for many years and previous work has included modelling of specific fault conditions using a macro-model. In this paper, the models are implemented in a more abstract form using analogue Hardware Description Languages (HDLs), including MAST, taking advantage of the ability to control the behaviour of the model using high-level fault condition states. The implementation method allows a range of fault conditions to be integrated without switching to a completely new model. The various transistor faults are categorised, and used to characterise the behaviour of the HDL models. Simulations compare the accuracy and speed of the transistor and behavioural level models under a set of representative fault conditions.

Keywords -Behavioural Fault Modelling, VHDL-AMS, MAST, Hardware Description languages, Simulation, Operational Amplifier

1 Introduction

It is becoming increasingly necessary to use mixed signal simulation to understand the behaviour of circuits under fault conditions. In order to practically implement simulation techniques it is necessary to implement realistic fault models, simulate large numbers of possible fault conditions in a reasonable time and test the resulting behaviour against the design specification.

If these three requirements are considered in reverse order, the first decision to make is what kind of testing approach to take? The two main types of approach are specification based and fault model based. In the specification based approach the circuit is tested against the specification [1] and a fault is deemed to have occurred only if the resulting measurement of performance is outside the specification range (e.g. rise time or bandwidth). Another approach is to build up a 'fault dictionary' of the standard faults characterised for each device or circuit [2-6]. In this case, the fault occurs when the model exhibits specific faulty behaviour that may still meet the specification. Using this approach, when the fault occurs, then the behaviour can be matched against the previously obtained fault types and immediately identified.

It is obvious from the requirements of the fault simulation approaches, especially the fault model based technique, that exhaustive simulations are required to identify the faulty behaviour. This requirement virtually mandates the intelligent use of behavioural modelling techniques to reduce the simulation times required [7]. A significant issue with the implementation and use of behavioural models in simulation for analogue integrated circuits is the matching of the device behaviour with the one found by transistor level simulation. Usually the transistor level simulation is carried out in a variety of SPICE simulators (e.g. HSPICE), and the question arises, how can the behavioural models be characterised easily and used in conjunction with these transistor level descriptions? Typical behavioural simulators, such as Saber, may use a proprietary language (MAST [8]). Despite the capability of the simulator at the behavioural level, the results of the transistor level simulations may not match those found by SPICE-like simulators, such as HPSICE, exactly (due to differences in the underlying transistor models and solution methods). The standard language IEEE 1076.1 (VHDL-AMS) [9] may also be used in simulators such as VeriasHDL or HAMSter, but the same problem arises that behavioural simulators are not renowned for their ability to deal adequately with transistor level simulations for large circuits. It is clear therefore, that checking is required at all stages to ensure that the behavioural models are consistent with the benchmark transistor level models. If this is done then minor differences between the implementations in different simulators can be minimized.

Implementing realistic fault models at different levels of abstraction is necessary to provide the required accuracy of the mixed level simulations. Using transistor level simulation models to establish the basic behaviour under a set of predefined fault conditions provides a baseline to which the behavioural model can be characterised. The choice can then be made as to which type of behavioural modelling approach is required. In this paper, the concentration is on the operational amplifiers, and there are two main approaches for behavioural modelling of these circuits; the macro-model and the equation based model. The standard Boyle macro-model [10] has been used for many years to behaviourally model opamps, but with the development of modern HDL based simulators, such as Saber, a more direct equation based implementation of

[1] This work was partly supported by a grant from the Engineering and Physical Sciences Research Council (United Kingdom) and Advanced Power Components (Rochester, United Kingdom).

opamp behaviour is possible. This paper deals with the equation-based approach to implement fault behavioural model using catastrophic and parametric structural faults.

The structure of the rest of the paper is as follows: First, transistor level opamp modelling is discussed. Then closed loop fault behavioural model is developed for a benchmark opamp circuit. Later, an open loop fault behavioural model is discussed. Finally, some conclusions are drawn.

2 Transistor Level Modelling

2.1 Introduction

While the simulation of devices at the transistor level is computationally very expensive, it is generally accepted that this provides a somewhat realistic and accurate result. It has been previously discussed in [5] and [7] how faster simulations can be carried out by using hierarchy and replacing some devices with behavioural models, without a severe penalty on the accuracy achieved. In this paper, the transistor level models are used as a benchmark to establish the behaviour of the device, and this is then used to characterise the behavioural model to the same level of accuracy, but with a much faster simulation.

2.2 Benchmark Operational Amplifier

To demonstrate the concepts used in this paper, the IEEE Mixed-Signal Benchmark opamp circuit has been used [11], the schematic for which is shown in figure 1. The SPICE netlist for this opamp is given in Figure 2, which also shows the Level 3 MOS transistor parameters used in the benchmark circuit.

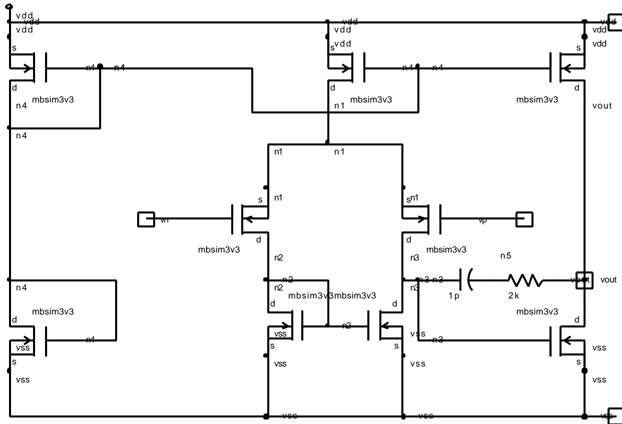

Figure 1: Transistor Level Operational Amplifier

```
.subckt OpAmpFaultFree VO VP VN Vdd Vss
Rc NET32 VO 2E3 M=1.0
Cc NET48 NET32 1E-12 M=1.0
M6 NET48 VP NET44 Vdd PMOS L=4E-6 W=30E-6 M=1.0
M3 NET35 VN NET44 Vdd PMOS L=4E-6 W=30E-6 M=1.0
M9 VO NET48 Vss Vss NMOS L=3E-6 W=154.2E-6 M=1.0
M4 NET35 NET35 Vss Vss NMOS L=4E-6 W=15E-6 M=1.0
M7 NET48 NET35 Vss Vss NMOS L=4E-6 W=15E-6 M=1.0
M2 NET54 NET54 Vss Vss NMOS L=32E-6 W=3E-6 M=1.0
M8 VO NET54 Vdd Vdd PMOS L=4E-6 W=200E-6 M=1.0
M1 NET54 NET54 Vdd Vdd PMOS L=4E-6 W=12E-6 M=1.0
M5 NET44 NET54 Vdd Vdd PMOS L=4E-6 W=30E-6 M=1.0
.MODEL   NMOS   NMOS   LMIN=1.2E-06   LMAX=1.5E-06
WMIN=2.0E-06 WMAX=500E-06 LEVEL=3
+VTO=.79  GAMMA=.38  PHI=.53  RD=63  RS=63  IS=1E-16
PB=.8 CGSO=1.973E-10
+CGDO=1.973E-10    RSH=45    CJ=0.00029    MJ=.486
CJSW=3.3E-10 MJSW=.33 JS=0.0001
+TOX=2.5E-08 NSUB=8.7E+15 NFS=8.2E+11 TPG=1 XJ=1E-
07 LD=7E-08 UO=577
+VMAX=150000 FC=.5 DELTA=.3551 THETA=0.046 ETA=.16
KAPPA=0.05
.MODEL    PMOS    PMOS    LMIN=1.2E-06    LMAX=100E-06
WMIN=2.0E-06 WMAX=500E-06 LEVEL=3
+VTO=-8.40000000E-01 GAMMA=.53 PHI=.58 RD=94 RS=94
IS=1E-16 PB=.8
+CGSO=3.284E-10 CGDO=3.284E-10 RSH=100 CJ=0.00041
MJ=.54 CJSW=3.4E-10 MJSW=.3
+JS=0.0001  TOX=2.5E-08  NSUB=1.75E+16  NFS=8.4E+11
TPG=1 XJ=0 LD=6E-08 UO=205
+VMAX=500000  FC=.5  DELTA=.4598  THETA=.14  ETA=.17
KAPPA=10
.ends OpAmpFaultFree
```

Figure 2: Transistor Level Operational Amplifier Netlist

2.3 Transistor Level Fault Free Behaviour

Using the transistor level operational amplifier model described, the behaviour of the device from the inputs to the output can be characterised in the inverting, non-inverting and unity gain configurations using the test circuits given in figures 3(a), (b), and (c), respectively.

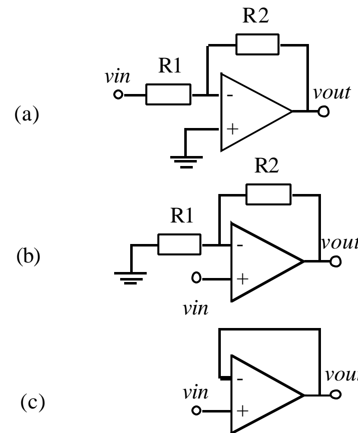

Figure 3: Closed Loop Test Circuits (a) Inverting, (b) Non-inverting, (c) Unity Gain

Using these test circuits, the fault free operational amplifier model was tested using Hspice and Pspice, with the input offset voltage and the output voltage measured in each case. The results of these simulations are shown in figures 4, 5, and, 6. The DC transfer analysis was used in this case as the faults could be classified using this aspect of the device behaviour alone. The DC transfer analysis in Saber allows the specification of the starting and finishing voltage values (-3V and +3V respectively), and a voltage step size (10mV in this study).

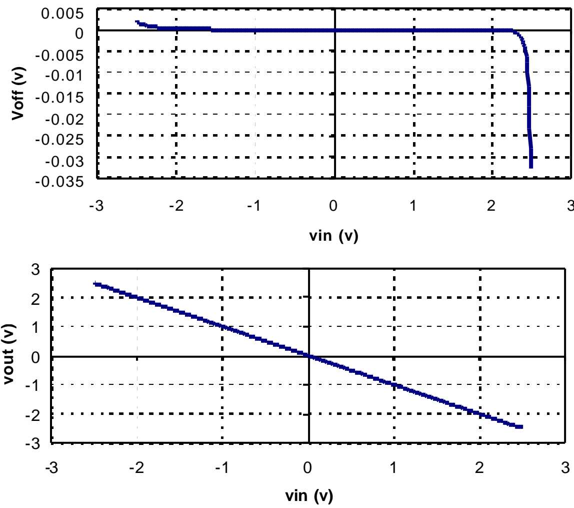

Figure 4: Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage

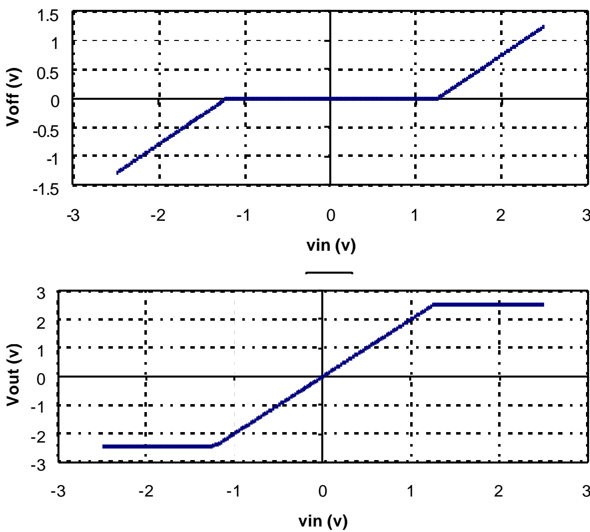

Figure 5: Non-Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage

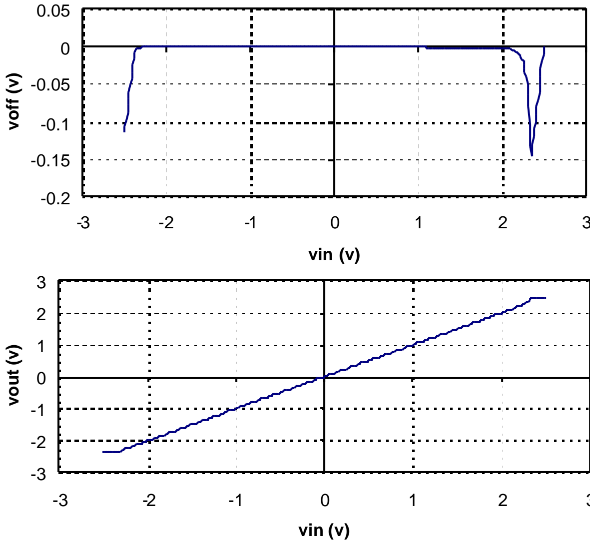

Figure 6: Unity Gain Amplifier Input Offset Voltage (vp-vn) and Output Voltage

2.4 Operational Amplifier Fault Behaviour

Recently work has been done to group the catastrophic and parametric faults that can occur in operational amplifiers by looking at the offset voltage at the inputs of the opamp, while carrying out DC sweep analysis [11]. Catastrophic faults are those that occur when an open or short circuit causes a complete failure in the operation of the device. Parametric faults, on the other hand, are variations in the MOS transistor channel lengths and widths, and threshold voltages, which cause a minor variation in the device's specification, such as gain and bandwidth. In this paper we are concentrating on the catastrophic faults, but the same models can be used to characterise the parametric faults as well.

The catastrophic faults for the opamp shown in Figure 1 can be categorised into four main types, type I (M5 Drain-Gate Short), type II (M7 Drain open), type III (M5 Drain to Source Short) and type IV (M5 Drain-Gate Short). Fault types I-III are for the inverting amplifier configuration, while the type IV fault applies to the non-inverting configuration. Figures 7-10 show DC transfer characteristics for four fault types.

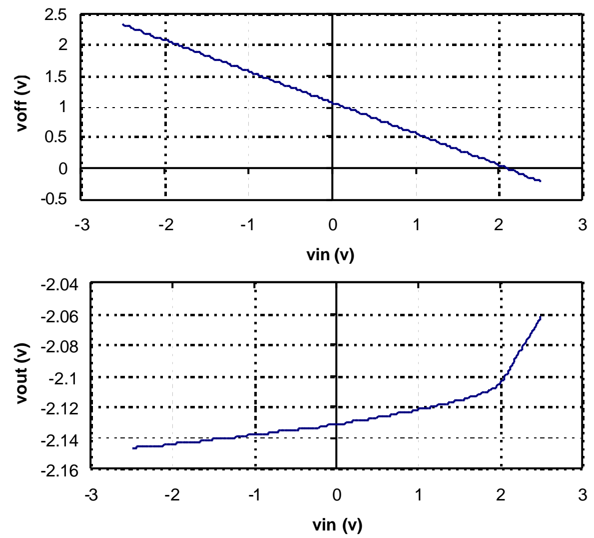

Figure 7: Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage with type I fault

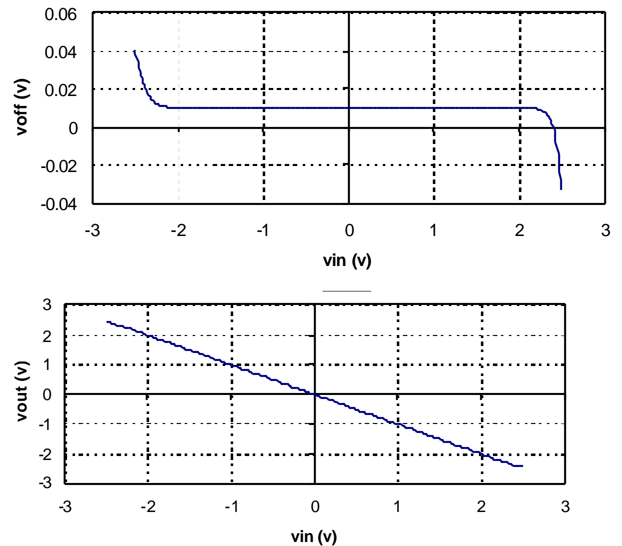

Figure 8: Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage with type II fault

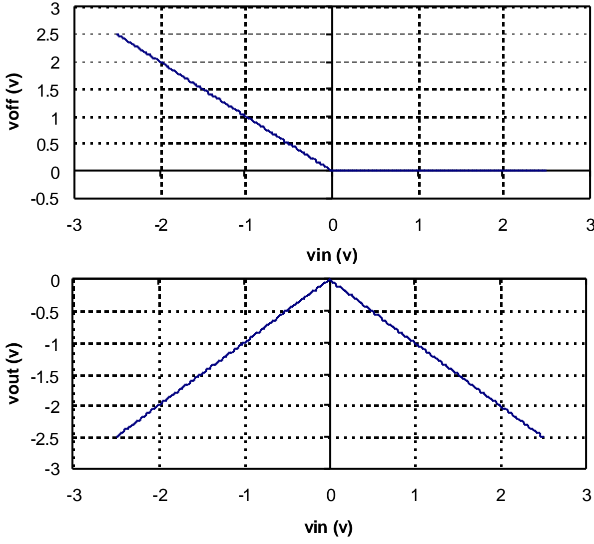

Figure 9: Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage with type III fault

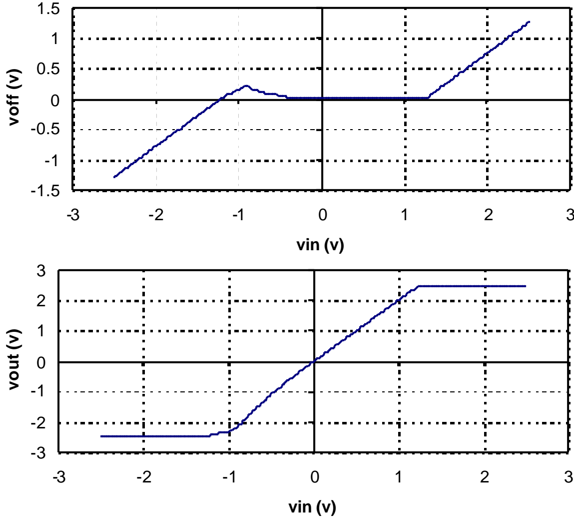

Figure 10: Non-Inverting Amplifier Input Offset Voltage (vp-vn) and Output Voltage with type IV fault

3 Closed Loop Behavioural Modelling

3.1 Close Loop Model Equation

With the complete transistor level modelling of the operational amplifier for the fault free and faults I-IV completed, this behaviour can be modelled behaviourally in a closed loop form. Change et al [11] provide a simple closed loop model of the form given in (1) which gives the input-output voltage relationship of the behavioural model.

$$V_{out} = A_{CL}\left[(1+m)V_{in} + k\right] \qquad (1)$$

where A_{CL} is the Closed Loop gain of the opamp, and m and k are parameters that characterise the non-ideal opamp effects, such as the limited output resistance, and the opamp's faulty behaviour for the closed loop configuration.

How the parameters m and k can be derived analytically from the parameters of the opamp is described in [7] and [11]. In this paper, the parameters were derived directly using the transistor level simulation results. For the fault free case, the parameters can be derived by inspection. For example, in the fault free inverting case, the gain is –1 (R_1 = R_2 = 1MegΩ), and therefore m=0 and k=0. For type I faults (a stuck at fault), if m=-1, then the value of the output voltage will simply be -k, where k is the magnitude of the stuck at voltage.

3.2 Proposed Closed-Loop Fault Behavioural Model

The model defined by equation (1) was implemented as a behavioural model using the MAST modelling language and simulated with the Saber simulator. The model listing is provided in Figure 11.

```
template opamp_behav2 vin voutgnd = a,m,k
electrical vin,vout,gnd
#...Operational Amplifier Parameters
number a=1
#...Fault Offset Voltage Parameters
number m=0
number k=0
{
#...Declarations
var i i
val v vo,vi,fo,voutcalc
#...Procedural Expressions
values {
        #...Terminal Voltages
        vo = v(vout) - v(gnd)
        vi = v(vin) - v(gnd)
        #...Fault Offset Voltage
        fo = m*vi + k
        voutcalc = a*(vi+fo)
        #...Supply Voltage Limit
        if (voutcalc > 2.5) voutcalc=2.5
        if (voutcalc< -2.5) voutcalc = -2.5
}
equations {
        #...Fundamental Equations
        i(vout->gnd) += i
        vo = voutcalc
        }
}
```

Figure 11: Closed Loop opamp MAST model

3.3 Testing the basic fault model

Using the same test benches as were used in the transistor level simulations, the behavioural model was tested in the fault free and each of the fault type cases with a DC transfer analysis. In the inverting fault free case, m=k=0 with the output voltage as shown in figure 12.

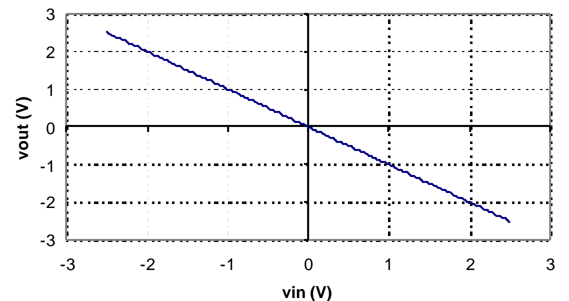

Figure 12: Inverting Amplifier Input Output Voltage for a fault free behavioural model

In the fault I case, the output voltage is a stuck at voltage, with the values -2.14 at -2.5V input to -2.11 at +2V. The region 2V to 2.5V is slightly steeper, with the variation -2.11 to -2.06 for the input range 2V to 2.5V. Therefore to

get a highly accurate mapping of output voltage behaviourally would require a PWL model. However, in this case the fact that the model is exhibiting a stuck at fault is adequate, and is accurate to within 5%. To get equation (1) to exhibit this behaviour, m=-1 (cancelling out the input voltage terms) and, k is just the value of stuck at voltage, which is probably best matched at vin=0, $vout$=-2.13V, therefore k=2.13. The Saber simulation result of the behavioural model for this fault type is given in Figure 13.

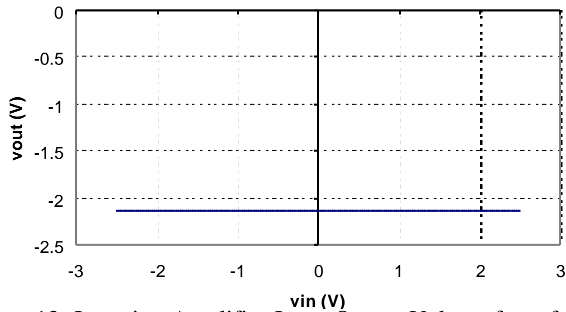

Figure 13: Inverting Amplifier Input Output Voltage for a fault I behavioural model

In the type II faults, the opamp works almost correctly, but the input offset is much higher than normal (10mV), causing a slight offset in the output. This manifests itself with a slight offset in the output voltage, leading to early saturation on one side of the output voltage swing. The basic behavioural model does not cope with this and as such must in fact include a limiting function to limit the output voltage to the supply rails (+/- 2.5V in this case). The resulting simulation of the output voltage is shown in figure 14.

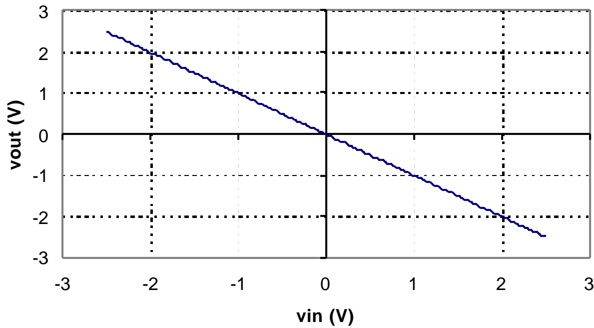

Figure 14: Inverting Amplifier Input Output Voltage for a fault II behavioural model

In the type III fault case, when the input voltage is greater than zero, the inverting opamp circuit works correctly, but when vin<0, the circuit behaviour turns into non-inverting. In terms of the equation (1), this implies that k=0 and that m is 0 for vin>0 and for vin<0, m=-2. This is summarised in equation (2).

$$vout = \left| \begin{array}{ll} A_{cl} vin & vin > 0 \\ A_{cl}(vin + m * vin) & vin < 0, m = -2 \end{array} \right. \quad (2)$$

This discontinuous behaviour cannot be modelled using the simple behavioural model previously given, and a modification was made in the model to include this change in behaviour as shown in Figure 15.

```
#...Fault Offset Voltage
if (vi <0) {
        fo = -2*vi
}
else {
        fo = 0
}
vout = a*(vi+fo)
```

Figure 15: Modification to the behavioural model for type III faults

Using this modified model, the resulting behaviour can be seen in figure 16.

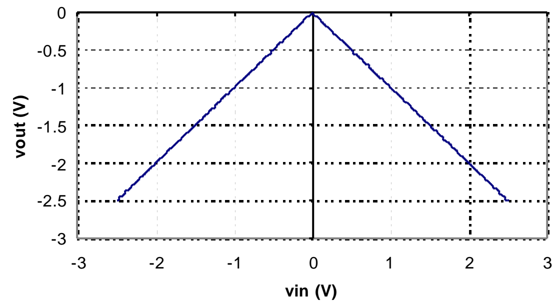

Figure 16: Inverting Amplifier Input Output Voltage for a fault III behavioural model

The same behavioural model can also be used for the non-inverting closed-loop case. In this configuration, the transistor level behaviour can be replicated using the parameters m=0 and k=0 with A_{CL}=2 ($A_{CL} = 1 + R_2 / R_1$ for the non-inverting opamp). The resulting simulated output voltage is shown in Figure 17.

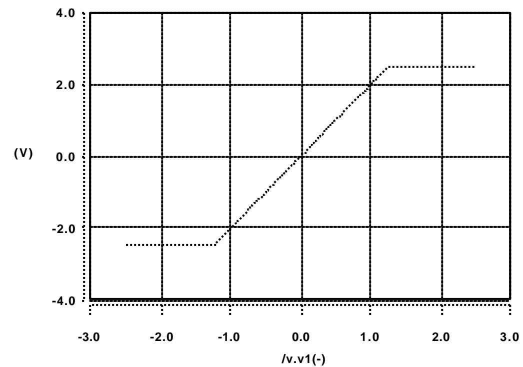

Figure 17: Non-Inverting Amplifier Input Output Voltage for a fault free behavioural model

On inspection of the transistor level results for the fault IV case, it is clear that a PWL approximation is required to provide a realistic match with a behavioural model. The closed loop model was therefore modified using a PWL voltage offset as shown in Figure 18.

```
#…Definition of the PWL structure
struc {
        number vi, vo
        }   pwlv[*]   =   [(-2.15,0.016),(-
1.3,0.02),(-1.13,-
0.111),(0.6,0.003),(2.5,0)]
… Existing Model Code
#… Calculation of the fault voltage
vos = pwl1(2,pwlv,vin)
```

Figure 18: Modification to the behavioural model for type IV faults

The resulting change in behaviour is subtle, but is a slight non-linearity introduced on the output voltage. The output voltage obtained using the behavioural simulation is given in Figure 19.

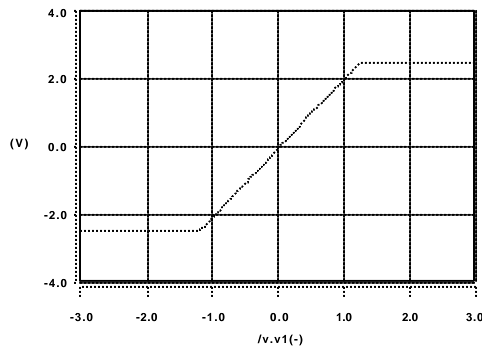

Figure 19: Non-Inverting Amplifier Input Output Voltage for a fault IV behavioural model

4 Open Loop Behavioural Modelling

One drawback with the closed loop model is the restriction on the topologies that can be simulated. If extra components are added into the feedback loop for example, then a complete re-characterisation is needed. To combat this, therefore, a modified fault offset voltage model was created that could be connected to the input of the opamp creating the required fault offset voltage, while allowing an arbirtrary connection to the opamp externally. The resulting mast model is given in figure 20.

```
template fos inp inm fosp=m,k,fault
electrical inp,inm,fosp
number m=0,k=0
enum { _fault1, _fault2, _fault3, _fault4, _none,
_specified } fault=_none
{
var i i
val v vin,vos
foreign pwl1

struc {
        number  vi, vo
        }  pwlv[*]  =  [(-2.15,0.016),(-1.3,0.02),(-
1.13,-0.111),(0.6,0.003),(2.5,0)]
#       }       pwlv[*]    =    [(-2.5,0),(-1.2,0),(-
0.9,0.2),(-0.2,0.01),(1.0,0),(2.5,0)]

#...Procedural Statements in this section
values {
        #...Calculate   input   voltage   vin   from
voltage on pins inp and inm
        vin = v(inp)- v(inm)
        if (fault == _fault1) {
                vos=-1.02*vin + -2.215
        }
        else if (fault == _fault2) {
                vos = -0.011
        }
        else if (fault == _fault3) {
                if (vin < 0 ) {
                        vos = 0
                }
                else {
                        vos = -2*vin
                }
        }
        else if (fault == _fault4 ) {
                vos = pwl1(2,pwlv,vin)
                vos = -0.5*vos
        }
        else if (fault == _specified) {
                vos = vin*m + k
        }
        else {
                vos = 0
        }
}

equations {
        i(fosp->inp) += i
        i : v(inp) - v(fosp) = vos

        }
}
```

Figure 20: General Purpose Opamp Fault Model

This model can then be connected in series with any behavioural opamp model with the option of either specifying specific fault types, or defining the *m* & *k* parameters directly. The advantage of using this type of approach becomes clear when more complex circuits are tested such as the IEEE Mixed-Signal Benchmark Biquad filter shown in Figure 21. Obviously, the opamps are not the simple buffers previously analysed using the closed loop model, and as such the open loop fault model becomes an ideal approach to simulating faults in this type of circuit. If this circuit is simulated in the fault free case the resulting output signal is given in Figure 22.

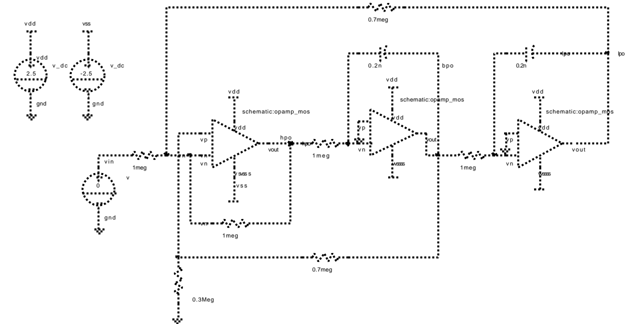

Figure 21: Biquad Filter Benchmark circuit

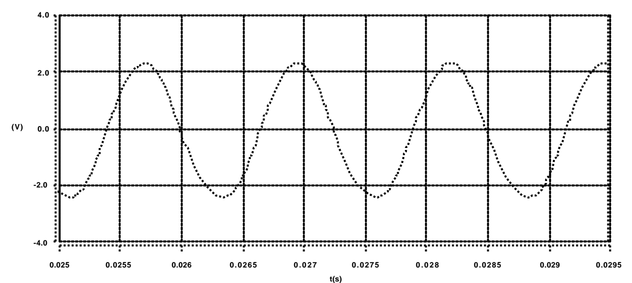

Figure 22: Biquad Filter fault free transient response

If the fault model for a type I fault is implemented in any of the opamps using the open loop behavioural model, then the output will be a stuck at voltage, as is shown in Figure 23.

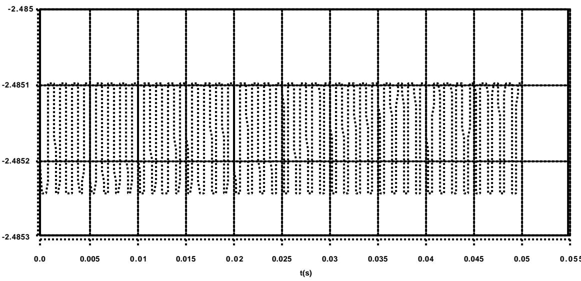

Figure 23: Biquad Filter fault I transient response

There is a slight ripple on the output, but essentially the output is stuck at -2.4V.

5 Summary of the results

It is clear from these results that there is a good correlation between the transistor level and behavioural level models. The real benefit for multiple simulations depends also on the simulation times in each case, and these are summarised in table 1. For each case a number of runs (3) was taken in each case and the average simulation time (CPU seconds) was recorded.

Circuit configuration	Transistor level Simulation Time (s)	Behavioural level Simulation Time (s)
Inverting Fault Free	1.88	0.1
Inverting Fault I	1.78	0.1
Inverting Fault II	1.81	0.1
Inverting Fault III	1.89	0.15
Non-Inverting Fault Free	1.89	0.1
Non-Inverting Fault IV	1.89	0.45
Biquad Filter (a)	11.1	2.35
Biquad Filter (b)	93.2	2.49

Table 1 : Comparison of Simulation Times[2]

In most of the opamp test cases, the speed up in using behavioural models was 18 times. This was slightly reduced in the Fault III and Fault IV models due to the extra solution time required to deal with the more complex PWL characteristic in the model. For the Biquad filter in case (a) the input voltage was half the supply voltage, and in case (b) the input was full scale. It is clear from the difference in simulation times that the relative merits of the MOS and Behavioural models may depend significantly on the operating region of the devices. If the devices are pushed into their non-linear regions, then the solution time may drastically increase as is the case here (x9). It is interesting to note that the behavioural model simulation time is almost exactly the same in both cases implying that the model is robust and can handle the limiting to the supply aspects without undue convergence difficulties.

6 Conclusions

In this paper, a method of implementing fault behavioural models for operational amplifiers has been presented. Previous work has been extended to cover both the open and closed loop configurations allowing greater flexibility in the application of the fault models in the general case. Results show a good correlation between transistor and behavioural models at all stages, with a corresponding improvement in simulation times.

7 References

[1] Voorakaranam R. and Chatterjee A., "Hierarchical specification driven analogue fault modeling for efficient fault simulation and diagnosis", *IEEE International Test Conference*, 1997, pp903-912

[2] Perkins A.J., Zwolinski M.J., Chalk C.D. & Wilkins B.R., "Fault Modeling and Simulation using VHDL-AMS", *Analogue Integrated Circuits and Signal Processing*, Vol. 16, 1998, pp141-155

[3] Pan C.Y. & Cheng K.T., "Fault Macromodeling for Analogue/Mixed Signal Circuits", *IEEE International Test Conference,* 1997

[4] Milor L. & Visvanathan V., "Detection of catastrophic faults in analogue integrated circuits", *IEEE Transactions on Computer Aided Design (CAD),* Vol. 8, No. 2, 1989, pp114-130

[5] Sounder T.M. & Stenbakken G.N., "A comprehensive approach for modeling and testing analogue and mixed signal devices", *Proceedings of International Test Conference*, 1990, pp169-176

[6] Nagi N., Chatterjee A., Balivida A., & Abraham J.A., "Fault based automatic test generator for linear analogue circuits", *Proceedings of International Conference on CAD*, 1993, pp88-91

[7] Kilic Y. & Zwolinski M., "Behavioural/Macro modelling to speed up analogueue fault simulation", ELECO 2001 (to be presented)

[8] MAST Reference Manual, Avant! Corporation , USA

[9] IEEE Standard 1076.1 Analogue and Mixed Signal Extensions to VHDL

[10] Boyle G.A., Pederson D.O., Cohn B.M. & Solomon J.E., "Macromodeling of Integrated Circuit Operational Amplifiers", *IEEE Journal of Solid State Circuits,* 1974, pp353-363

[11] Chang Y., Lee C.L., Chen J.E., & SU C., "A Behaviour Level Fault Model for the Closed-Loop Operational Amplifier", *Journal of Information Science and Engineering,* 2000, Vol. 16, pp751-766

[2] All the simulation times in table 1 were obtained using Saber 5.1 running on a Celeron 500MHz PC with Windows NT.