Automated Analog Design

Part II

A symbolic analysis approach

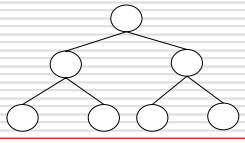
Objectives

- Some notes about last my presentation
- Abstract
- Review related works
- Introduction of Symbolic analysis
- Proposal for automated analog design
- Examples
- Difficulties and future goals
- Conclusions

Notes about last my presentation

Genetic algorithms

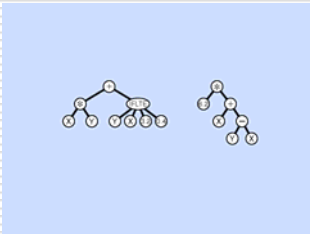
- A genetic algorithm transforms a population of individual objects, each with an associated value of fitness into a new generation of the population, using the principles of survival and reproduction of the fitness and analog of natural occurred operation crossover and mutation
- In order to use this algorithmic approach we map object representation with labeled tree .
  Ex.: object is a formula (X+0.98)*(Y-0.5)



Notes about last my presentation

Genetic algorithms

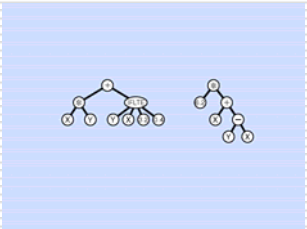
Crossover
Two randomly chosen nodes exchange the corresponding subtrees



Notes about last my presentation

Genetic algorithms

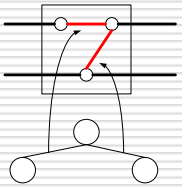
Mutation
At randomly chosen node grows randomly chosen function from the set of all available functions



Notes about last my presentation

Genetic algorithms

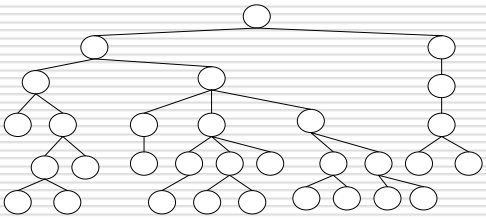
- Design analog filter
  The algorithm starts with embryo structure – two modifiable wires Z_0,Z_1
- The two branches below "list" node is associated with Z_0,Z_1
- The two branches below "list" node is associated with Z_0,Z_1



Notes about last my presentation

Circuit constructing program tree

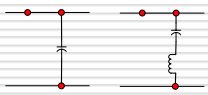
Offspring in II generation



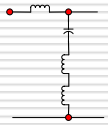
Notes about last my presentation

Genetic algorithms

- Circuit , synthesizing, by traversing nodes from the top to the bottom



- Circuit, corresponding to the offspring



- Functions:
  - C,L,R Capacitor, inductance, resistor
  - Flip – reverse voltage polarity
  - Series – following functions are build series structure
  - End – not modifiable node
  - Cut – modifiable wire will be deleted
  - Two_groung – divide modifiable wires in 2 parts and connect middle node to ground
  - Pair_connect – connect two nodes

Notes about last my presentation

the inaccurate genome representation

-Genome – a collection of genes , representing parameters of the problem to be optimized
- Example of genome:

Abstract

Some arguments supporting automated circuit analog design

- Market requirement for short design cycle
- Decreasing products life cycle
- The supply and demand growth of solutions, rather than a parts.

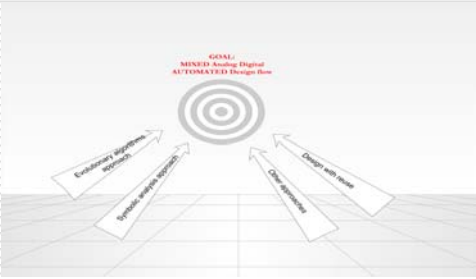
13

Abstract

- The background of the all difficulties
- Possible achievement …
  - Optimization aspect
  - Automated sizing aspect
  - VHDL aspect
- What is not done yet
  - Fully automated mixed analog/digital circuit design engine

14

Abstract



Abstract

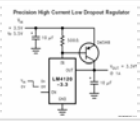
- What is the real situation
- Design by reuse
  - Based on existing IP
  - Usually reference design is first optimal one
  - Possibly to be not optimal solution
  - Quick design
  - Virtually no optimization is needed

Design by reuse – example 1

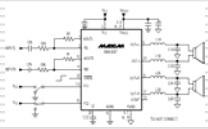
Most leading IT companies use referent design for some products

Design by reuse – example 2

-Precision Micropower Low Dropout Voltage Reference LM420 (National)
- 9 referent circuits (www.national.com)



-Mono/Stereo 2W (Class-D) Audio Power Amplifier MAX4295
- 1 referent circuit



Review related works

- Neolinear
  - Several tools for creating IP in analog/mixed/Soc design
  - NeoCircuit is used to automatically size, bias and verify analog circuits
  - NeoCircuit-RF is used to automatically size, bias and verify RF circuits including LNAs, mixers, etc.
  - NeoIP is a library of analog cells (VCO, bandgap, opamps, etc.) fully constrained for use in your NeoCircuit and NeoCell design environment.
  - NeoCell, for automatic analog place and route, is distributed and supported by Cadence Design Systems.

Review related works

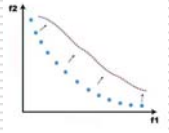
- **Analog design automation Ltd**
  - **Commercial software for analog/digital/mixed design**
  - **Operate with existing IP database in electronic design**
  - **Provide fast design/evaluation on custom specifications**
  - **Evolutionary algorithms makes partial design and optimization until the goal is reached.**
  - **Strongly dependent on existing database**

Review related works

- **Design overview**

In any circuit, there are many possible optimal designs, each representing the best tradeoffs in design goals (e.g., area vs. speed; speed vs. power, etc.).

The system engineer has to find the one design from many optimized candidates that best meets the overall design objectives

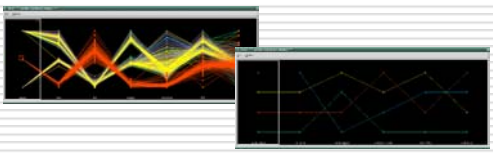


Review related works

- **Creative Genius® tool**
- searches for optimized designs using evolutionary algorithms.
- It considers all design variables and design objectives in its search of the design space.
- CG provides a quick analysis, producing several results from simulation.
- The values of the performance measurements are expressed as **design goals** (speed, area, power, etc.)
- Performed tasks: maximize, minimize or match the design goal to a specific value

Review related works

- Output visualization makes easy choosing best design according initial constrains (example Low THD / F)
- Even the we design an amplifier, circuit topology is different for different goals



Review related works

- **Existing solutions – an overview**

BENEFITS

- Decreasing (up to 10 times) design cycle
- If the certain design cannot be evaluated , searching space is bounded by output set of optimized simulation-validated circuits

DRAWBACKS

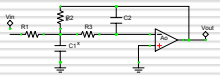
- Depending on the IP database of initial topologies.
- Not fully automated.
- Essentially without guarantee to find a solution.

Bandgap reference	Manual		Creative Genius	
	Metric	Meet Spec?	Metric	Meet Spec?
Vout Nominal, V	1.26274	No	1.23305	Yes
Vout spread vs Temperature, V	0.03658	No	0.00059	Yes
TK_max, ppm/C	403.894	No	24.3301	Yes
Vout overall deviation, V	0.02546	No	0.00259	Yes
Gain_at_1V_out	194.319	No	8740.29	Yes
Gain_at_1500mV_out	282.817	No	12153.2	Yes
Cfgain, dB	41.3671	No	119.533	Yes
UGBW, Hz	5591710	Yes	3757830	Yes
PM_deg	72.6793	Yes	88.6515	Yes
Isupply, A	0.0008715	Yes	0.0005256	Yes
Inp_offset voltage_V	0.005118	Yes	0	Yes
Area, um^2	1954.5	Yes	2475.74	Yes
Vout_V	1.27784	Yes	1.23309	Yes
Vout_peak_V	3.47994	Yes	3.51049	Yes
Settling_time_s	3.39E-08	Yes	2.47E-08	Yes
Max_gain_dB	88.9036	Yes	163.918	Yes
Gain_at_500mV output, dB	59.6472	No	140.341	Yes
Gain_at_750mV output, dB	41.3063	No	119.522	Yes

Introduction of Symbolic analysis

- Benefits
  - explicit circuit representation
  - Analytical qualitative analysis
  - opportunity to look at the circuits as a set of subcircuits.
- Difficulties
  - huge set of equation describing analog processes
  - it is hard to be fully analyzed medium size IC
  - in many cases, circuits with more than 100 elements, the relatively small error in process equation is growing at the output.
- GOAL: Finding and canceling all insignificant terms

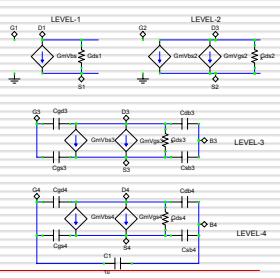
Introduction of Symbolic analysis

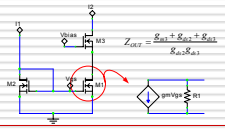


- **Types of analyses (Gain Vout/Vin):**
- Fully symbolic $\frac{G_1 G_2}{s_2(C_1C_2(1+\frac{1}{A_0})+s(C_2(G_1+G_2+G_3)(1+\frac{1}{A_0})+\frac{G_1G_2}{A_0})+G_1G_3(1+\frac{1}{A_0})+\frac{G_1G_2}{A_0}}$
- Simplified symbolic $-\frac{G_1 G_2}{s_2C_1C_2+sC_2(G_1+G_2+G_3)+G_1G_3}$
- Semisimbolic $\omega^2=(\frac{A_0+3}{A_0+1})(0.5.10^9)$

Introduction of Symbolic analysis

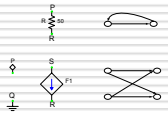
- MOS Transistor – Small signal models
- The model choice depends on the desired analysis, objective function, circuit size etc.
- All the models are linearized





Introduction of Symbolic analysis

Computational techniques:



Tree enumeration methods:
- Handle only small circuits
- Doesn't produce term cancellation

Flow graph (topological) method:
- Mason signal FG – weighted signal graph of simultaneous linear equations
- Each node is a summer $X_j = \sum W_{jk} X_j$
- Transfer function between two nodes is

$$\frac{X_j}{X_i} = \frac{1}{\Delta}\sum_k P_k \Delta_k$$

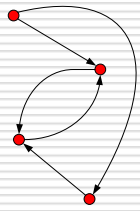
Δ-> sum of all lups+(sum of all 2nd lups)-(sum of all 3nd lups)...

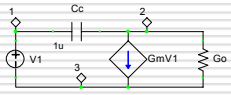
Δ_k-> Δ touching k node

P_k-> sum of all loops at k node

Introduction of Symbolic analysis

Example Mason method



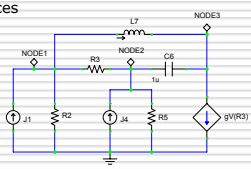


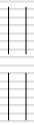
$$\frac{V_2}{V_1} = \frac{1}{1+\frac{sC}{g_0}}\left[\frac{g_m}{g_0} + \frac{sC}{g_0}\right] = \frac{sC - g_m}{sC + g_0}$$

Introduction of Symbolic analysis

Computational techniques:

Matrix based methods: fully symbolic equations, obtained directly from the circuit. **Modified Nodal Analysis MNA** formalize the problem in the form YV=J, Y admittance matrix, V voltage, J current sources





Linearized circuit equation

$$\begin{pmatrix} G_2 + G_3 + \frac{1}{sL_7} & -G_3 & -\frac{1}{sL_7} \\ -G_3 & G_3 + G_5 - sG_6 & -sG_6 \\ g_8 + \frac{1}{sL_7} & -(g_8 + sG_6) & \frac{1}{sL_7} + sG_6 \end{pmatrix} \times V = J$$

Gain (V3/V2) obtained applying Cramer rule

$$\frac{V_3}{V_2} = \frac{(g_8 - \frac{1}{sL_7})(-J_4G_3 - J_1G_5 - J_5sC_6) + (g_8 + sC_6)(J_4G_2 + J_4G_3 + \frac{J_4}{sL_7} + J_1G_5)}{(g_8 + sC_6)(-sC6 + \frac{J_4}{sL_7}) + (sC_6 + \frac{1}{sL_7})(J_1G_3 + J_1G_5 + J_5sC_6 + J_5G_3)}$$

Introduction of Symbolic analysis

Hierarchical Symbolic analysis

- Overview
  - The main idea is to partition the circuit into a number of smaller circuits with already compiled parameter.
  - Two types of blocks – inner and outer (leafs, terminal).
  - Time complexity $O(n^2/p)$
- Terminal block analysis
  - Use traditional MNA approach
  - Reducing MNA to RMNA by reducing all internal variables
- Middle block analysis
  - Recursively combining pair of blocks with at least 2 common nodes, to produce leaf nodes.

Introduction of Symbolic analysis

Hierarchical Symbolic analysis

- Example of middle block analysis

$$Y_1 = \begin{pmatrix} G_1 & -G_1 A \\ 0 & sC_2 + G_2 - sC_2 A \end{pmatrix}$$

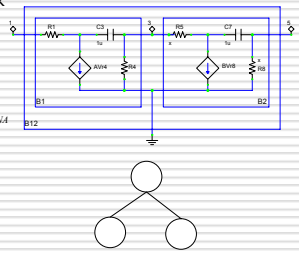
$$Y_1 = \begin{pmatrix} G_3 & -G_3 B \\ 0 & sC_4 + G_4 - sC_4 B \end{pmatrix}$$

$$Y_{12} = \begin{pmatrix} G_1 & -G_1 A & a_{13} \\ 0 & sC_2 + G_4 - sC_2 A + G_5 & -G_5 B \\ 0 & 0 & sC_4 + G_4 - sC_4 B \end{pmatrix} \Rightarrow MNA$$

$$Y_{12} = \begin{pmatrix} G_1 & -G_1 A - G_5 B \\ 0 & sC_2 + G_4 - sC_2 A + G_5 \\ 0 & sC_4 + G_4 - sC_4 B \end{pmatrix} \Rightarrow RMNA$$

$$Y_{12} \times \begin{bmatrix} V^{12} \\ I^{12} \end{bmatrix} = \begin{bmatrix} J^{12} \\ E^{12} \end{bmatrix}$$

$$Y_{12} \times \begin{bmatrix} V^{12} I^{12} \\ I^{12} I^{12} \end{bmatrix} = \begin{bmatrix} J^1 \\ J^2 \end{bmatrix}$$



Introduction of Symbolic analysis

- Basic definition:
  - **Topology graph G (N, B, R)** N (nodes), B (branches), R (incident relation)
  - **Structure S(G, F),** F(B->E, each branch is associated circuit element)
  - **Circuit C(S,P)** P(E->R), each element is takes real numerical assignment
  - **Objective function D** (determine circuit performance with respect design specifications)
  - **Error function Er**, Er=|D-Do|)
  - **Element cost Cs** individual element cost function Cs : E->cost
  - **Cost function Ks, Kc**, for structure Ks : S->R or for circuit Kc : C->cost

Introduction of Symbolic analysis

- Problem formalization

1. Minimization of cost function with objective minimal error function

$$\min_{s \in S} Ks(S) \quad \text{or} \quad \min_{c \in C} Kc(C)$$

2. Optimization Objective function with constraints imposed on cost function

$$\min_{s \in S} Es(S) \quad or \quad \min_{c \in C} Ec(C)$$

3. Global optimization

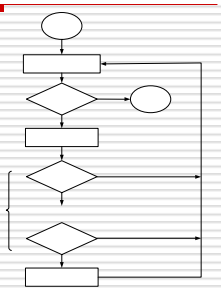
$$\min_{s \in S}(\alpha \times Er(S) + \beta \times Ks(S))$$

$$\min_{c \in C}(\alpha \times Er(C) + \beta \times Ks(C))$$

Introduction of Symbolic analysis

Sequential structure generation schemes

- P1: Definition of a class possible topologies
- P2: Exhaustive generation of all possible topologies
- P3: For all structures generation of a symbolic characteristic function of a circuit



Proposal for automated analog design

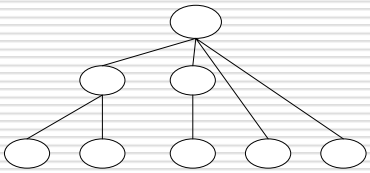
- By using symbolic analysis can be build of database with compiled (analyzed) base circuits. Example: differential amplifier, 2 MOS current mirror, etc.
- On the level circuit can be design correct mapping between topology and real (existing) elements.
- Additional parameter describing the certain behavior of each topology is assigned. Example (diff. amplifier – AF, IF; common drain amplifier – AF, IF, RF). There is no need to analyze diff. amplifier for RF.
- Each topology is presented with the full admittance matrix and additional parameters set for terms cancellation. Example Cgb is important in HF analysis. It's similar to chose the level of transistor simulation.

Proposal for automated analog design

- Design specification **Task(S, Err, Class)**, where S input-output signal bandwidth, Err – maximal error function, Class – some ID to the type design has to be performed.
  Ex. Task((100W, THD<0.03%, AF stereo amplifier)
  - Find set of partitions in all structures corresponding to **Class** $\{D_1, D_2, \ldots, D_n\}$
  - Find all alternative substitutions for D_i $\{D_{i1}, D_{i2} \ldots D_{ik}\}$
  - Perform symbolic analysis for each compatible in D_i with constrain min (Er (D_i)), min (cost) and |S'-S|<Err

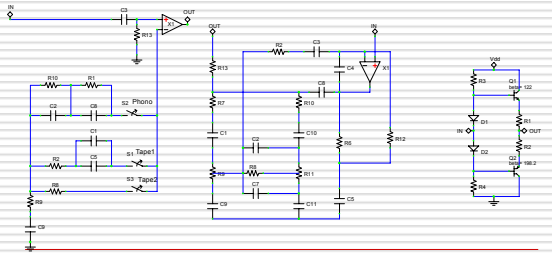
Proposal for automated analog design

- Example Goal : Task((100W, THD<0.03%, AF stereo amplifier.
  - 100W output power requires at least two amplifier stages
  - AF requires bias control
  - THD<0.03% require low noise pre-amplifier OPAMP

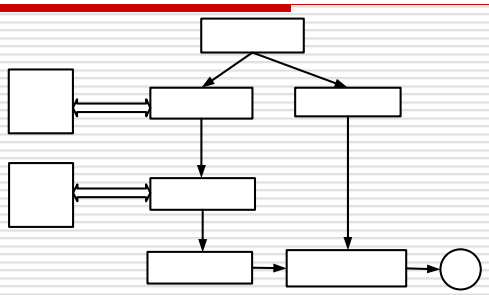


Proposal for automated analog design

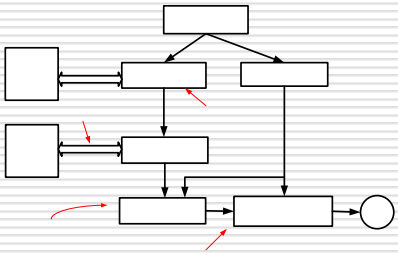
- Example of topologies maintaining problem constrains



Proposal for automated analog design



Proposal for automated analog design



Proposal for automated analog design

- The parameter "index of applicability (AP)" can save time for unnecessary computation, by assignment approximate value for quantitative estimation how much the structure will fit solving a particular problem. **Example Low noise can be maintain with tube amplifier stages. So AP for tube structures will point somewhere in high level of "low noise requirement"**
- The overall problem can be defined as a constrain satisfaction problem. Additional constrain will be index of applicability – analyzing the structures with greater AP. In this case only the cost constrain will be validated

Proposal for automated analog design

- What are the benefits?
  - Missing GA for optimization
  - Strong analytical approach finding a solution
  - Possibly working in fully automated mode
  - Intended to work much faster.
- What will be difficult?
  - Probably not so fast in optimization.
  - It will never come up with new solution.
  - Possibly hitches in the symbolic analysis part – cannot find solution due to memory overflow

er

OP AMP

Correcti ons

AB AMP

Difficulties and future goals

- Difficulties:
  - How to find more cancel-free terms?
  - I'm not sure that all specification parameters have explicit analytical presentation(example noise has approximate formulas ?)
  - I don't know yet what will be the time complexity.
  - I'm still looking for acceptable ontology
  - I don't know how to compare symbolic expressions

Difficulties and future goals

- Future goals
  - I'll look for specific algorithms in symbolic analysis . There is an alternative computation using BDD (ZDD)
  - I'll try make some GA programs, just to estimate their usefulness.
  - I'll start working on expert system only in a amplifiers domain. (Lisp, CLISP?)

- Hello (Alex):

  Good to hear from you.

  There is no current GPPS code at the present time because we swtiched computers shortly after the 1999 book. We may get back to GPPS later this year.

  Best of luck with your work.

  John R. Koza

  Consulting Professor
  Biomedical Informatics
  Department of Medicine
  Medical School Office Building (MC 5479)
  Stanford University
  Stanford, California 94305-5479