FASY: A Fuzzy-logic based tool for Analog SYnthesis

A.Torralba, *Member*, IEEE, J.Chávez and L.G.Franquelo, *Member*, IEEE

March 1995

Abstract

A CAD tool for analog circuit synthesis is presented. This tool, called FASY, uses fuzzy–logic based reasoning to select one topology among a fixed set of alternatives. For the selected topology, a two–phase optimizer sizes all elements to satisfy the performance constrains minimizing a cost function. In FASY, the decision rules used in the topology selection process are introduced by an expert designer or automatically generated by means of a learning process that uses the optimizer mentioned above. The capability of learning topology selection rules by experience, is unique in FASY. Practical examples demonstrate the tool ability of this tool to learn topology selection rules and to synthesize analog cells with different circuit topologies.

1 Introduction

Although a great deal of effort has been recently devoted to the development of CAD tools for the automatic design of integrated circuits, most of this effort has been directed to digital design. The design of the analog part of a circuit is still a time–consuming task. This is due to the complex relations that exist between design parameters and circuit performances [1].

The typical starting point of an analog design, is a circuit topology and a set of parameters obtained from the designer's experience. Then, several simulations and redesigns are carried out until an acceptable circuit is achieved. This redesign process is a very time–consuming task. In addition, a designer's experience is required to interpret the results obtained from the simulation and to modify the design parameters in the proper direction.

This process has been of practical use while the design of analog cells has been restricted to a reduced number of experts. The increasing demand of application–specific ICs (ASICs) that are entirely or partly analogic, makes necessary the development of CAD tools, which can aid in decreasing the design time, allowing non–experts to design complex analog circuits.

Circuit design can be considered as a two–step process: the designer first selects a circuit topology among a set of fixed alternatives and then optimizes the values of circuit parameters to meet a set of specifications.

Existing CAD tools have been mainly focused on the parameter optimization phase. Early optimization tools used a step–by–step procedure previously determined by an expert designer [3]–[4]. The final design may be considered as a starting point for manual parameter adjustment. The following tools converted the optimization process to a minimization problem by the formulation of a cost function. Different techniques have been used to minimize the cost function. An early review of these techniques can be found in [5]. Some recent tools will now briefly be reviewed.

The OPASYN system [6] uses the steepest descendent method. Due to the greedy nature of the algorithm, it will normally find a local minimum, so that several trials with randomly generated starting points will be required to assure a good design. The method of *feasible directions* is used in DELIGHT.SPICE [7]–[8]. It uses an enhanced version of the simulation program SPICE, to compute circuit performances and sensitivities

that are employed during the optimization process. OAC [9] has a two–phase optimizer: during the first phase, a simple optimization procedure is applied to an analytical model of the circuit. The second phase is based on the Levenbarg–Marquardt optimization method, and SPICE is used to compute the cost function. In [10] NPSOL, a software package for sequential quadratic programming is used. Statistical techniques, such as simulated annealing, can avoid being trapped in a local minimum, so that a solution close to the global minimum will be found at the expense of a high computational cost [11]–[12]. More recent work can be found in [13]–[16].

The topology selection process usually relies on a designer's experience. This is a type of knowledge which is very difficult to capture, as it is entirely qualitative and even changes from one designer to another. Existing tools dealing with the topology selection process, use artificial intelligence techniques. In [17] a rule–based system synthesizes operational amplifiers from easier components. OPASYN [6] uses heuristic rules to select one among a fixed set of alternatives. BLADES [18] uses a divide and conquer strategy, partitioning the circuit design in smaller tasks which are solved by subcircuit design modules. OASYS [19] and An_Com [20] are frameworks for analog circuit synthesis that rely on a hierarchical representation of circuits. In each hierarchical level, a two–step topology selection and translation process is carried out.

Fuzzy–logic based systems have been specially proposed to deal with "uncertain" information and have proved to be very efficient in capturing human expertise [21]–[22]. This makes them good candidates to aid in circuit design. The application of fuzzy logic to the design of ICs has begun to produce excellent results [14],[23].

This paper presents FASY, a design system for analog circuits that uses fuzzy logic in the topology selection process. Afterwards, a two–phase optimizer sizes all the components of the selected topology to minimize a user–defined cost function. Now, FASY is capable of synthesizing a variety of CMOS operational amplifiers.

The paper is organized as follows. Section II describes the general architecture of FASY. The topology selection process is detailed in Section III. Section IV describes the device–sizing optimizer used in FASY and presents some practical designs. Finally, Section V presents some concluding remarks and advances

future research.

2 System architecture

The block diagram of FASY is shown in figure 1. Two data flows are pointed out in this figure. In the top–down flow (called *design flow*), FASY acts as a design tool for analog cells. In the bottom-up flow (called *learning flow*), the topology selection rules are learnt from the experience gained with the optimizer of FASY. In a design with FASY, performance specifications are defined by the designer. Table I shows an example of the specification set for operational amplifiers. The topology selection module uses fuzzy decision rules to select the circuit topologies that best accommodate the set of performance specifications. Selected topologies are displayed along with a grade from 0 to 100, indicating their relative suitability to performance specifications. The designer chooses one of them or allows FASY to choose the highest rated topology. Then, the selected topology is given to the circuit parameter optimizer. If the final design is accepted, it is stored in the FASY data base. The experience gained with these successful designs can be used to modify the topology decision rules. This process is called *learning rules process* and can be automatically made by means of a backpropagation method that will be later described.

3 Topology selection

The process of choosing one topology among a set of fixed circuit topologies is called *design style selection* [24]. This process directly follows the standard manual design. Designers typically look for a proven technology that is capable of meeting performance specifications. Only in the case that none of the topologies can meet them, the designer starts the much more difficult task of designing a new topology.

In the topology selection process, FASY uses a rule based system based on fuzzy logic. Since its introduction in the sixties by L.A.Zadeh [25], fuzzy logic has been applied to different engineering fields, such as decision–making, control of industrial processes, pattern recognition and classification [21]–[22],[26]. Fuzzy logic has been specially proposed to deal with "uncertain" information and provides an effective means of

capturing the approximate inexact nature of human reasoning. Fuzzy logic variables are defined by means of "non–precise" linguistic terms, such as

x is *large* or *very large*

The process of extracting conclusions from a set of fuzzy antecedents is called *fuzzy inference*. Fuzzy inference is based on a set of decision rules whose antecedents and consequents are linguistic terms.

IF a *large* gain and a *small* area is required
THEN the suitability of topology 1 is *large*

The fuzzy system used in FASY comprises four principal components:

- A fuzzification interface that converts input data into suitable linguistic values which may be viewed as fuzzy sets. Fuzzy sets can be seen as a generalization of binary logic, where the binary values are substituted by a measure of truthfulness (membership grade) by means of a number in the range [0,1]. FASY employs triangular shaped membership functions to define the linguistic terms of the input variables and singletons to define the linguistic terms of the output variable (figure 3).

- A knowledge base that contains the definition of linguistic terms and a set of fuzzy rules which are in the form of *if then* rules. A fuzzy control rule, such as *if (x is A_i and y is B_i) then (z is C_i)* is implemented by a *fuzzy implication* R_i which is defined as follows:

$$\mu_{R_i} = [\mu_{A_i} \ \ and \ \ \mu_{B_i}] \ \ \rightarrow \ \ \mu_{C_i}$$

- The decision–making logic which employs fuzzy implication and the rules of inference in fuzzy logic to obtain a fuzzy output. Fuzzy implication in FASY uses the product operation rule. The connective

and is implemented by means of the algebraic product. Fuzzy rules are combined by means of the sentence connective *also* which is implemented in FASY by means of the algebraic addition. With this selection of parameters of the fuzzy system, an expression for rule i implication is given by:

$$\mu_{C_i'} = (\prod_{antecedents} \mu_{A_i}(x_i) \ ) \ \mu_{C_i} = \phi_i \ \mu_{C_i} \tag{1}$$

Figure 4 depicts a graphic interpretation of the inference process.

- A defuzzification interface which yields a nonfuzzy action from the inferred fuzzy output. Defuzzification is carried out through the centroid method, which generates the center of gravity of the membership function of the output set. As the membership functions which define the linguistic terms of the output variable are singletons, the center of gravity of the inferred fuzzy set can be obtained by means of the following expression:

$$y = \frac{\sum_1^m \phi_i w_i}{\sum_1^m \phi_i} \tag{2}$$

The knowledge base in FASY can be directly obtained from an expert designer or automatically generated from the experience gained with earlier designs using the parameter optimizer that will be described in the following section. Table II resumes a set of decision rules entered by a human expert when a large active area is allowed. Following these rules, the three basic topologies are adequate when small dc gains and small unity gain frequencies are specified. For small and medium dc gains, the Basic Two Stage (BTS) op.amp. is preferred, while the Folded Cascode OTA op.amp. dominates in the case of large gains and frequencies. Note that these rules are defined using the natural language used by expert designers. Using the rules given and the inference process described above the decision surfaces of figure 5a are obtained. These figures indicate the relative suitability of each topology to performance specifications, and are used in the topology decision process. Membership functions of figure 6 have been used to obtain the surfaces of figure 5. This example has been deliberately made simple so that the results can be represented as simple decision surfaces. FASY takes into accounts dc gain, bandwidth, active area, load, and also power consumption in

the topology decision process.

Sometimes the topology decision rules cannot be acquired due to the lack of experience (e.g., in the case of new technology) or due to the existence of conflicting specifications. In those cases, FASY is able to automatically generate the decision rules by means of neuro–fuzzy techniques.

The automatic rules generation process carries out the following steps. The specifications space is clustered in an adequate number of cells, each one representing a set of user specifications. In the case of only two specifications, it leads to rectangular bi–dimensional cells. In the case of n specifications, it leads to n–dimensional cells. For each cell and each possible topology, an optimization process is carried–out using the FASY optimizer. The final value of the cost function is considered as a figure of merit of the related topology with this set of user specifications. It is convenient to repeat the optimization process several times, starting with randomly generated initial values of the system variables to obtain a result independent of the optimization procedure. Our experience indicates that only five trials give good enough results. Using this method, the decision surfaces of figure 5b have been obtained. When compared with the decision surfaces of figure 5a some interesting conclusions can be drawn. The general behaviour of these figures are similar, indicating that the experienced designer had a good qualitative feel for topology selection. There is a difference, whereas figure 5a represents the approximate reasoning of a human expert, figure 5b has been obtained as a consequence of an optimization process, and represents precise knowledge of circuit behaviour. Each surface in figure 5b was built from the results obtained with 100 designs. Technology files were taken from a commercial 1.5 μm CMOS process. It took approximately six hours on a 96 MIPs workstation to obtain the decision surfaces of figure 5b. Whereas it seems to be a time–consuming task, this process only needs to be carried out once for a given technology.

These surfaces can be stored as fuzzy decision rules by means of a learning process based on the backpropagation algorithm, first introduced in [27] for a class of neural networks, and later adapted for fuzzy systems. A short description of the learning process follows (a more detailed description can be found elsewhere, e.g. in [29], [28]).

1. Initiate the parameters a_{ij}, b_{ij} and w_j (figure 3). In the absence of any "a priori" knowledge, they can

be randomly chosen.

2. Choose an input–output pair ($\mathbf{x}$,y^d) from the list to be learnt. $\mathbf{x}$ is the input vector and y^d the desired output.

3. Compute intermediate values ϕ_j and the output y corresponding to the input vector $\mathbf{x}$ (forward process).

4. Update parameters according to the following equations (backward process):

$$\Delta w_j = \eta_w \ [y^d - y] \ \phi_j \tag{3}$$

$$\Delta a_{ij} = \eta_a \ [y^d - y] \ \frac{\partial y}{\partial a_{ij}} \tag{4}$$

$$\Delta b_{ij} = \eta_b \ [y^d - y] \ \frac{\partial y}{\partial b_{ij}} \tag{5}$$

5. Return to step 2 until

$$Error = \frac{1}{2} \sum_{learning \ \ points} (y^d - y)^2 \ < \ \epsilon \tag{6}$$

The derivatives $\partial y/\partial a_{ij}$ and $\partial y/\partial b_{ij}$ can be obtained using the chain rule. η_w, η_a and η_b are learning rate parameters.

Storing decision surfaces as fuzzy rules exhibits some advantages over a tabular representation. First of all, only a few decision rules are required for surface representation, saving memory. Secondly, fuzzy logic is a structured way for knowledge representation and can be used by expert designers in future designs, and also in the evaluation of new incoming technologies. Finally, storing decision surfaces as fuzzy rules, allows the system to inherit some properties of fuzzy systems such as generalization, that is, the ability to obtain good responses when faced with new problems. After the learning process, the decision surfaces do not significantly differ from the surfaces of figure 5b. Figure 7 shows the decision surfaces obtained with the proposed method in the case of a small active area (figure 6a) and in the case of a large active area (figure 6b). Comparing these figures it can be observed the evolution of the decision surfaces when the specified active area changes.

As knowledge is represented by means of the exact shape of a set of membership functions and fuzzy rules, changing from one technology to another, only requires replacing them with a new set of membership functions and fuzzy rules, maintaining the structure of the decision making process.

Note that the technique used in FASY for the topology selection of operational amplifiers can also be used for other analog cells, and in the selection step of each level of a hierarchical design tool, such as OASYS. As an example, figure 8 shows the decision surfaces obtained with the proposed method for the selection among two topologies of CMOS current mirrors (figure 9), a frequent decision for an analog designer. In this example, decision is made depending on the specified output resistance (R_o) and the output voltage V_{MIN}. Other factors, such as active area and are also taken into account in the selection process. Once again, note that the surfaces in figure 8 have been obtained as a consequence of an optimization process, and represents precise knowledge of the circuits behavior. These surfaces are stored as fuzzy rules in the data base of FASY and table III shows some examples of the selection carried out by FASY.

Using fuzzy logic for topology selection, and the ability to learn decision rules in an automatic way, are characteristics of FASY that are believed to be unique when it is compared with other existing analog CAD tools.

4 Design Optimization

The optimization phase is aimed at obtaining the size of circuit devices which minimizes a cost function, which depends on user specifications. In FASY the objective function defined in [9] has been selected.

$$f_{obj} = \frac{1}{2}\sum_{i=1}^{N_{esp}} f_i^2 \tag{7}$$

$$f_i = W_i\left(\sqrt{Err_i^2 + \frac{1}{W_i^2}} + Sign_i \cdot Err_i\right) \tag{8}$$

$$Err_i = \frac{P_i(\vec{x})}{P_i^{esp}} - 1 \tag{9}$$

where :

$\vec{x}$ represents the design parameter vector.

$P_i(\vec{x})$ is the i-th performance parameter.

P_i^{esp} is the i-th design specification.

W_i is the i-th design specification weighting factor (default value is 1).

$Sign_i$ is set to $+1$ (-1) when $P_i(\vec{x})$ is required to be less (greater) than the P_i^{esp}.

This objective function gives a high penalty to those specifications that are not met. On the other hand, there is a slight reward to those specifications that are "over–attained", showing the preference for better performances. The exact expression of the objective function is not of critical importance to FASY. Other different types of objective functions can be used, such as the one used in OPASYN which combines quadratic and exponential terms.

Due to the complex dependencies that exist between design parameters and circuit performances, only statistical methods, such as simulated annealing, can avoid being trapped in a local minimum [30]–[31]. In addition, an accurate estimation of the circuits performance requires the use of an analog circuit simulator like SPICE [32] in each annealing movement, leading to excessively large computation times. The experience gained using SPICE in each movement of an algorithm that uses the annealing strategy proposed in [33], took more than six hours on a 96 MIPs workstation, to optimize the simple BTS op.amp. of figure 2a.

Although several techniques have been proposed to speed–up the convergence of the simulated annealing, the overall computational time is far from being of practical use. The use of simple analytical models of circuits and devices can produce performance estimation precise enough, as is shown in [10]. These analytical models can be found elsewhere [34]–[36]. The same simulated annealing algorithm mentioned above, requires only a few minutes in the same optimization problem, when these simple analytical models are used instead of SPICE to compute circuit performances. To make the final results more similar to actual circuit performance, a second optimization phase is carried out in FASY that uses SPICE in each movement. It uses the results of the first optimization phase as a starting point, and the well–known Fletcher–Powell

conjugate gradient method [37] to obtain the final result.

The following paragraphs describe the simulated annealing technique employed in the first optimization phase of FASY. This technique is an adaption to the continuous space of the method first proposed in [33]. In each movement, system variables are modified in the following way:

$$\omega_{k+1}[j] = \omega_k[j] + coef \cdot rnd_k[j] \tag{10}$$

where:

$\omega_k[j]$ is the j-th a–dimensional system variable, in the k-th movement

$rnd_k[j]$ is a normalized random number, later described

$coef$ is a heuristic dumping factor defined as:

$$coef = \begin{cases} 2 * coef, & \text{if } objdf > max\_var, \\ coef/8, & \text{otherwise.} \end{cases} \tag{11}$$

and:

max_var is the maximum difference of the cost function in an accepted movement

$objdf = mean - min\_abs$

$mean$ is the mean value of the cost function in the previous annealing temperature

min_abs is the absolute minimum value reached until now.

$coef$ limits the move range when the annealing process proceeds. The initial temperature is determined by a set of initial movements with infinite temperature. The criterion in determining the equilibrium in each temperature, the strategy for temperature decrement and the stopping condition have been taken from [33]. The parameters of this annealing process are based on statistical quantities and are problem independent.

The random term in (10) is given by the expression

$$rnd_k[j] = (1 - tc) \cdot unrnd_k[j] \quad - \quad tc \cdot grad_k[j] \tag{12}$$

where $unrnd_k[j]$ is a normalized, uniformly distributed random number. Equation (12) also uses the normalized gradient of the objective function:

$$grad_k[j] = normalized \left(\frac{\partial f_{obj}}{\partial x[j]} \right)_{x_k[j]}$$

The new factor tc is in the range [0,0.8]. It makes the movements biased against the gradient direction. Its initial value is 0 and increases with temperature.

$$tc = \frac{1}{1 + temp/K} \tag{13}$$

The value of K determines the starting point of the gradient term. The method proposed here can be converted to simulated annealing by setting K to 0. On the other hand, by setting K to ∞, it is transformed to a steepest descendent method. In this case K has been set to 1, though it is not a critical value and good results are obtained within a wide range of values of K. To avoid an excessively greedy algorithm, tc is upper–bounded to 0.8.

The proposed method includes the gradient of the objective function in the random movement of the annealing process. Qualitatively, it performs as simulated annealing at high temperatures, and performs as a steepest descendent method near the frozen condition, avoiding the excessively time–consuming movements that are characteristic of the simulated annealing near the frozen condition. As a result, the number of movements are largely reduced while maintaining good results. Figure 10a shows the evolution of the cost function with the proposed method and figure 10b, with simulated annealing ($tc = 0$). Both figures were obtained during the optimization of an OTA circuit. A considerable reduction in the number of movements and in the total time can be observed. FASY uses the proposed method (i.e. $K = 1$) when the

first derivatives of the cost function with respect to design parameters are known. Otherwise, it switches to simulated annealing ($K = 0$).

The solution obtained at the end of this process is considered as a starting point for a second optimization phase, that uses a standard conjugate gradient algorithm. Circuit performances in each step are computed using SPICE. Table IV shows some results obtained after the two–phase optimization process, and the circuit topologies shown in figure 2. The grades given by the topology selection module are also shown, as well as the total time required for the optimization process. The good choice made by the topology selection module can be observed.

5 Conclusions and future research.

This paper presents FASY, a tool for automatic design of analog circuits. FASY exhibits some innovative features with respect to other existing analog tools. First of all, the topology selection process uses a rule–based system that relies on fuzzy–logic. In this process, different topologies are graded depending on the design specifications. The decision rules set can be introduced by an expert designer, or automatically obtained by means of a learning package, which uses the experience gained with early designs. Device sizing in FASY is a two–step process. In the first step, simple analytical models of devices and circuits, are used to obtain a solution that is near an absolute minimum of the cost function. In the second phase, a standard conjugate gradient algorithm, which uses SPICE to compute circuit performances, is used to obtain the final design, starting from the final point of the first optimization step. Presently a prototype implementation of FASY is capable of designing a variety of CMOS operational amplifiers. To complete the design process, work on an automatic layout tool is currently in progress.

Acknowledgment

The authors would like to thank Prof. J.Aracil for providing guidance in fuzzy logic. Thanks are also due to Miss Belinda M. for her assistance.

References

[1] M.Ismail, J.Franca, Eds. "Introduction to Analog VLSI Design Automation". *Kluwer Academic Publishers*, 1990.

[2] L.R.Carley, R.A.Rutenbar. "How to automate analog IC design". *IEEE Spectrum*, Aug. 1988.

[3] P.E.Allen, E.R.Macaluso. "AIDE2: An automated analog IC design system". *Proc. IEEE Custom Integrated Circuits Conf.*, 1985.

[4] M.G.R.Degrauwe, O.Nys, E.Dijkstra, J.Rijmenants, S.Bitz, B.L.A.G.Goffart, E.A.Vittoz, S. Cserveny, C.Meixenberger, C.V. det Stappen, H.J.Oguey. "IDAC: An Interactive Design Tool for Analog CMOS Circuits". *IEEE J. Solid–State Circuits*, vol. SC–22, no. 6, December 1987, pp. 1106–1117.

[5] R.K.Brayton, G.D.Hachtel, A.Sangiovanni–Vincentelli. "A survey of optimization techniques for integrated circuit design". *Proc. IEEE*, vol. 69, pp. 1334–1362, Oct. 1981.

[6] H.Y.Koh, C.H.Séquin, P.R.Gray. "OPASYN: A Compiler for CMOS Operational Amplifiers". *IEEE Trans. Computer Aided Des.*, vol. CAD–9, no. 2, February 1990, pp. 113–125.

[7] W.Nye, D.C.Riley, A.Sangiovanni–Vincentelli, André L.Tits. "DELIGHT.SPICE: An Optimization–Based System for the Design of Integrated Circuits". *IEEE Trans. Computer Aided Des.*, vol. CAD–7, no. 4, April 1988, pp. 501–519.

[8] J.M.Shy, A.Sangiovanni–Vincentelli. "ECTASY: A new environment for IC design optimization". *Proc. ICCAD 1988*, pp. 484–487.

[9] H.Odonera, H.Kanbara, K.Tamaru. "Operational–Amplifier Compilation with Performance Optimization". *IEEE J. Solid–State Circuits*, vol. SC–25, no. 2, April 1990, pp. 466–473.

[10] P.C.Maulik, L.R.Carley, D.J.Allstot. "Sizing of Cel-level Analog Circuits using Constrained Optimization Techniques". *IEEE J. Solid–State Circuits*, vol. 28, no. 3, March 1993, pp. 233-241.

[11] G.G.E.Gielen, H.C.C.Walscharts, W.M.C.Sansen. "Analog Circuit Design Optimization Based on Symbolic Simulation and Simulated Annealing". *IEEE J. Solid–State Circuits*, vol. SC–25, no. 3, June 1990, pp. 707–713.

[12] F.Medeiro, R.Domínguez–Castro, A. Rodríguez, J.L.Huertas. "A Prototype Tool for Optimum Analog Sizing Using Simulated Annealing". *Proc. of the IEEE Int. Symp. on CAS*, 1992.

[13] E.S.Ochotta, R.A.Rutenbar, L.R.Carley. "Equation–free synthesis of high–performance linear analog circuits". *Proc. 1992 Brown/MIT Conf. Advanced Research in VLSI and Parallel Systems*, Cambridge, MA:MIT Press 1992, pp. 129–143.

[14] B.R.S.Rodrigues, M.A.Styblinski. "Adaptive Multi–Objective Fuzzy Optimization for Circuit Design". *Proc. of the IEEE Int. Symp. on CAS*, 1993, pp. 1813–1816.

[15] J.Chavez, M.A.Aguirre, A.Torralba. "Analog Design Automation. A Case Study". *Proc. of the IEEE Int. Symp. on CAS*, 1993, pp. 2083–2085.

[16] N.S.Nagarj. "A New Optimizer for Performance Optimization of Integrated Circuits by Device Sizing". *Proc. of the IEEE Int. Symp. on CAS*, 1993, pp. 2102–2105.

[17] R.J.Bowman, D.J.Lane. "A knowledge–based system for analog integrated circuit design". *Proc. IEEE Int. Conf. Computer–Aided Design*, 1985.

[18] F.El–Turkey, E.E.Perry. "BLADES: An Artificial Intelligence Approach to Analog Circuit Design". *IEEE Trans. Computer Aided Des.*, vol. CAD–8, no. 6, June 1989, pp. 680– 692.

[19] R.Harjani, R.A.Rutenbar, L.R.Carley. "OASYS: A Framework for Analog Circuit Synthesis". *IEEE Trans. on Computer Aided Des.*, vol. CAD–8, no. 12, December 1989, pp. 1247–1266.

[20] E.Berkcan, M.d'Abreu, W.Laughton. "Analog compilation based on successive decomposition". *Proc. 1988 ACM/IEEE Design Automation Conf.*, June 1987.

[21] D.Dubois, H.Prade. *Fuzzy Sets and Systems: Theory and Applications*. Academic Press, San Diego, 1980.

[22] H.J.Zimmerman. *Fuzzy Sets, Decision Making and Expert Systems*. Boston: Kluwer, 1985.

[23] M.Hashizume et al. "A Parameter Adjustment Method for Analog Circuits Based on Convex Fuzzy Decision using Constrains Satisfactory Level". *Proc. of ICCD*, 1990, pp. 24–28.

[24] D.Thomas. "The automatic synthesis of digital systems". *Proc. IEEE*, vol. 69, Oct. 1981.

[25] L.A.Zadeh. "Fuzzy Sets". *Informat. Control*, vol. 8, pp. 338–353, 1965.

[26] M.Sugeno, Ed. *Industrial Applications of Fuzzy Control*, North–Holland, 1985.

[27] D.E.Rumelhart, G.E.Hinton, R.J.Williams. "Learning representation by back–propagation error". *Nature*, vol. 323, pp. 533–536, 1986.

[28] D.Driankov, H.Hellendoorn, and M.Reinfrank. *An Introduction to Fuzzy Control*: Springer–Verlag, N.Y., 1993.

[29] Chi–Cheng Jou. "Supervised learning in fuzzy systems: algorithms and computational capabilities". *Proc. of the 2nd IEEE Int. Conf. on Fuzzy Systems*, March 1993, pp. 1–6.

[30] S-Kirkpatrick, M.Vecchi. "Optimization by Simulated Annealing". *Science*, vol. 220, pp. 671–680. May 1983.

[31] R.A.Rutenbar. "Simulated Annealing Algorithms: An Overview". *IEEE Circuits and Devices Magazine*, vol. , pp. 19–26, Jan. 1988.

[32] L.W.Nagel. "SPICE2: A computer program to simulate semiconductor circuits". Memo No. ERL–M520, Electronics Research Laboratory, University of California, Berkeley CA, May 1985.

[33] M.D.Huang, F.Romeo, A.Sangiovanni–Vincentelli. "An Efficient General Cooling Schedule for Simulated Annealing". *Proc. of Int. Conf. on CAD*, 1986, pp. 381–384.

[34] P.R.Gray, R.G.Meyer. *Analysis and Design of Analog Integrated Circuits*. John Wiley, 1984.

[35] R.Gregorian, G.C.Temes. *Analog MOS Integrated Circuits for Signal Processing*. New York:Wiley, 1986.

[36] R.L.Geiger, P.E.Allen, N.R.Strader. *VLSI Design Techniques for Analog and Digital Circuits*. McGraw–Hill, 1990.

[37] D.G.Luenberger. *Linear and Nonlinear Programming, 2nd. ed.*. Addison-Wesley Publishing Inc., Mass. 1984.

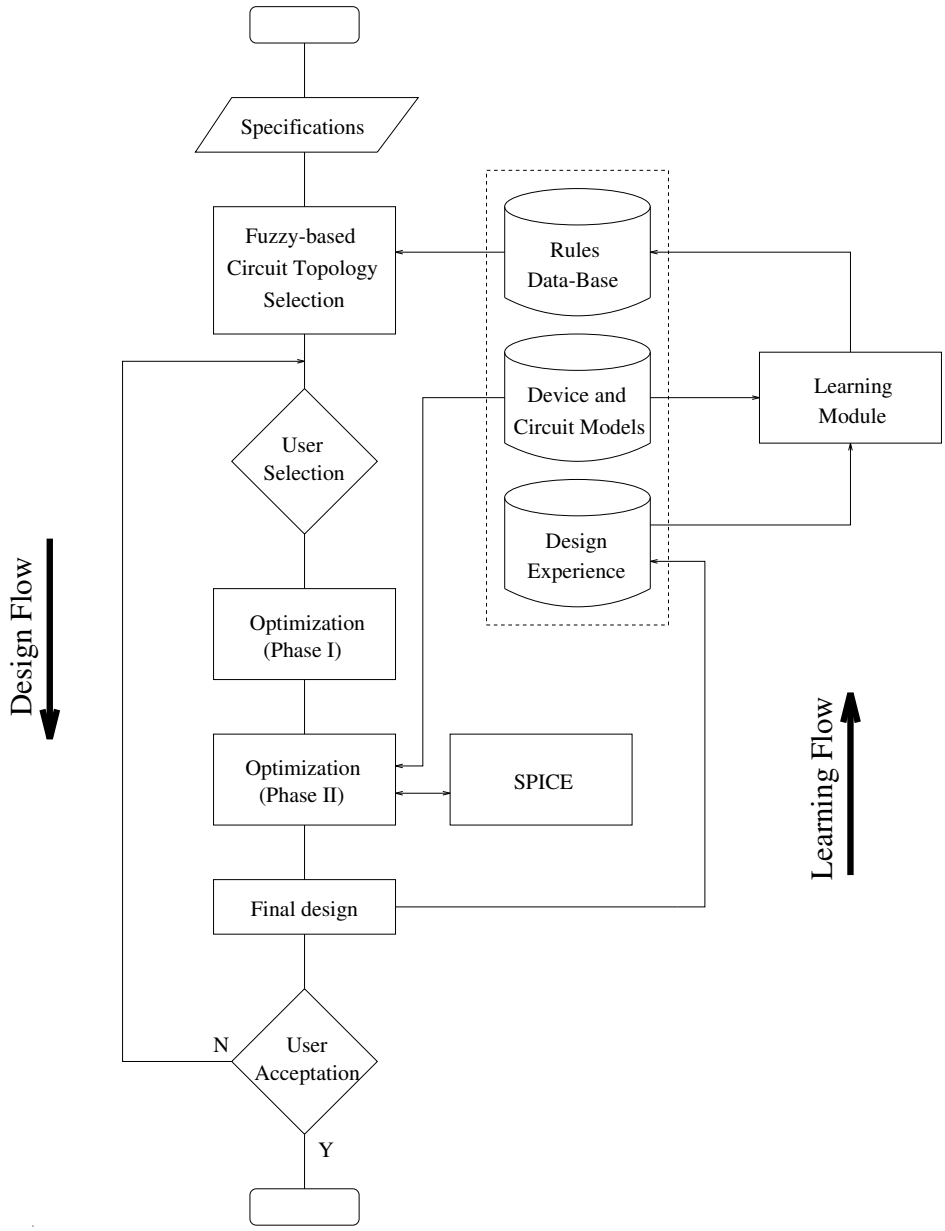


Figure 1: Block diagram of FASY.

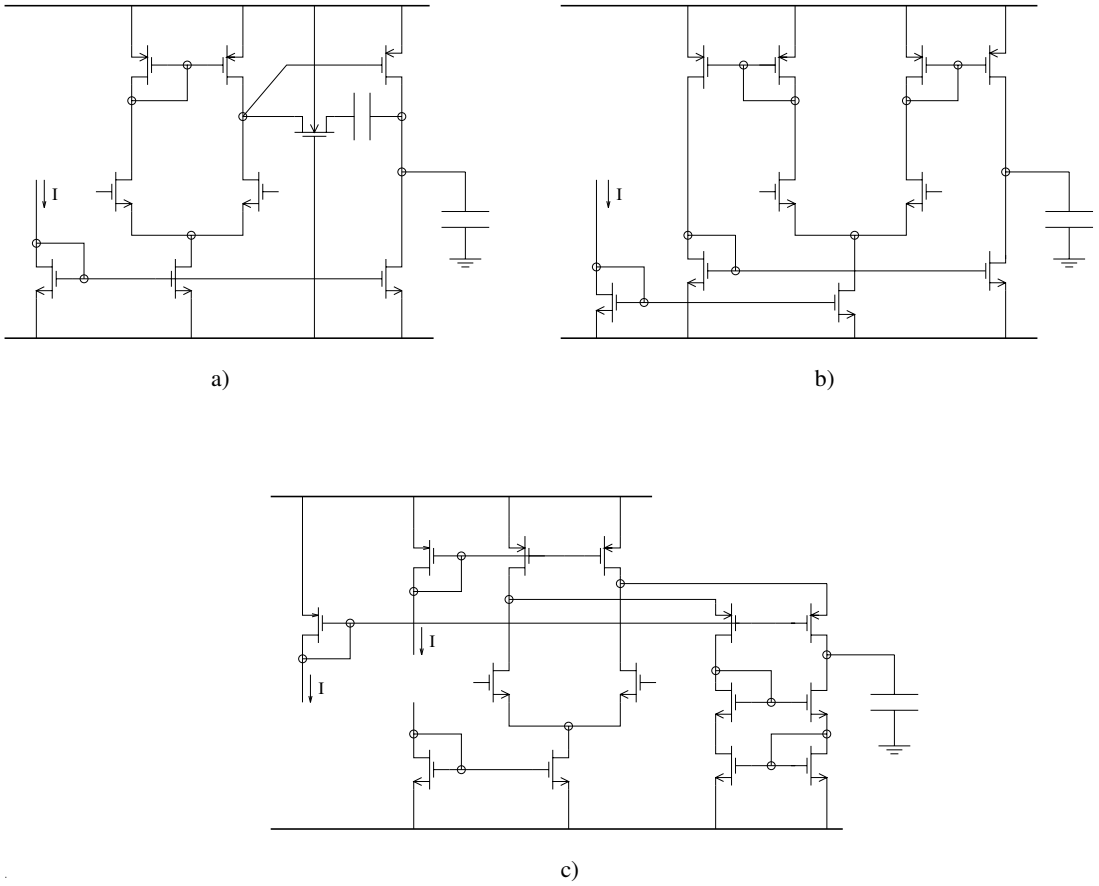


Figure 2: Basic op.amp. topologies: a) Basic Two Stage (BTS) op.amp., b) Output Transconductance Amplifier (OTA) and c) Folded Cascode OTA.

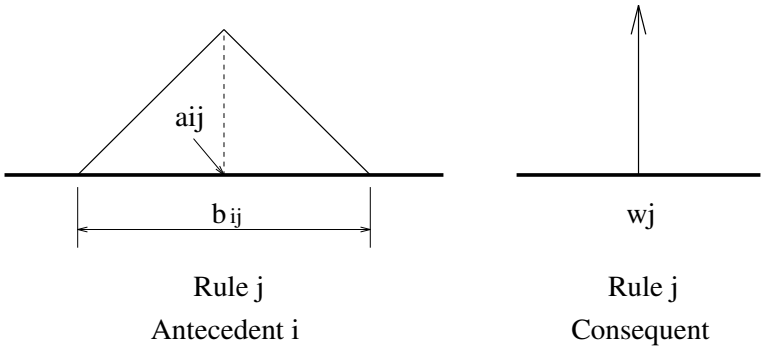


Figure 3: Membership functions (μ).

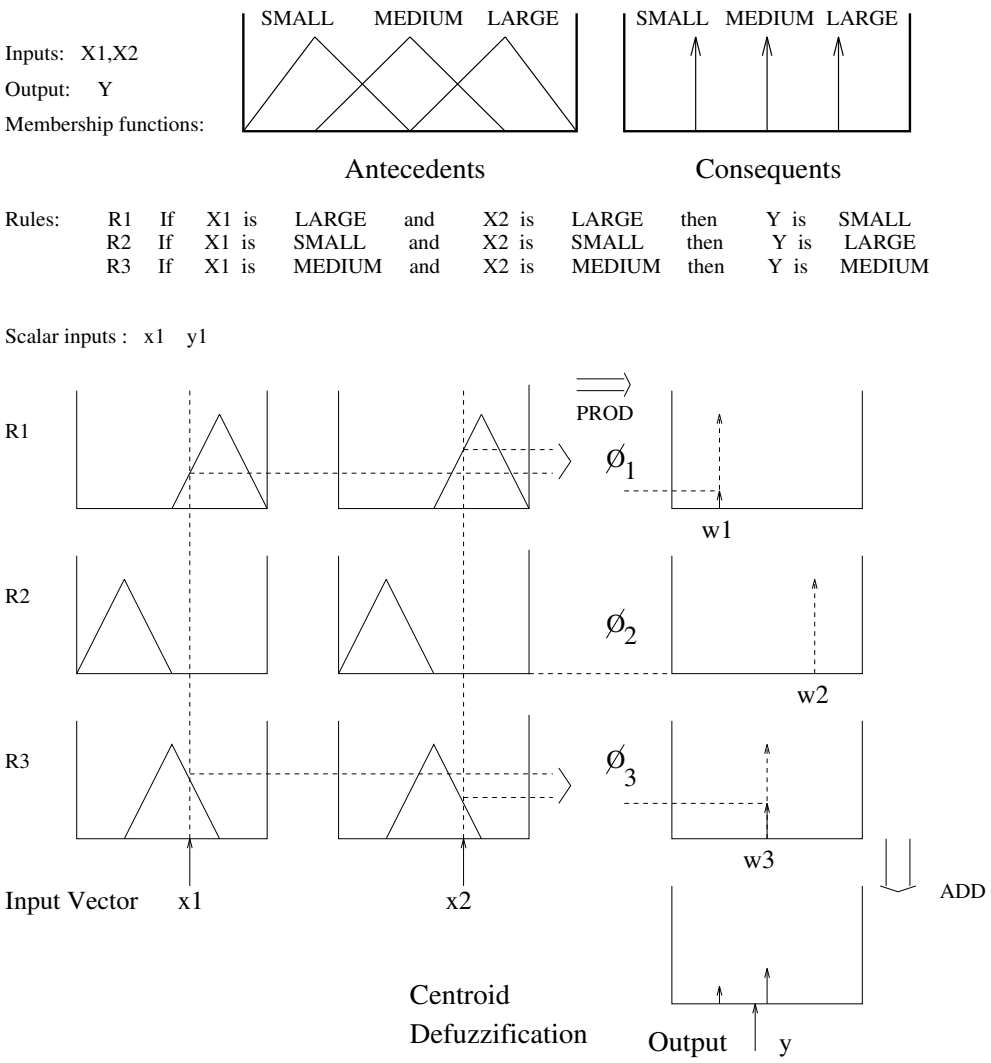
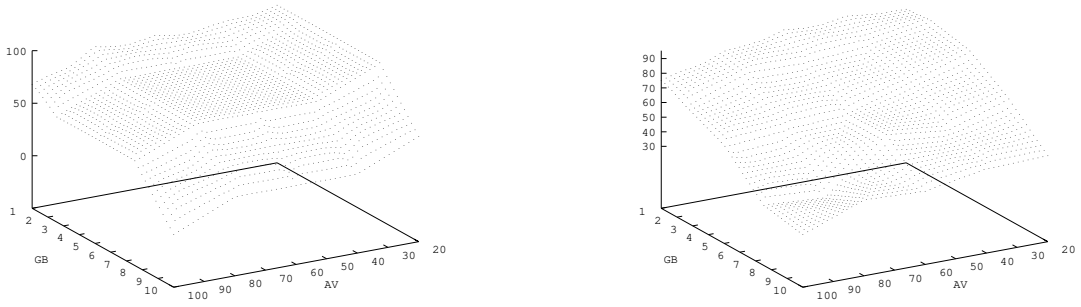
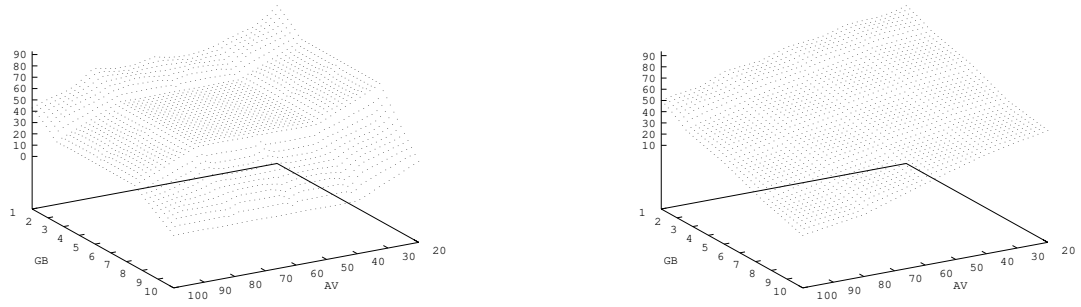


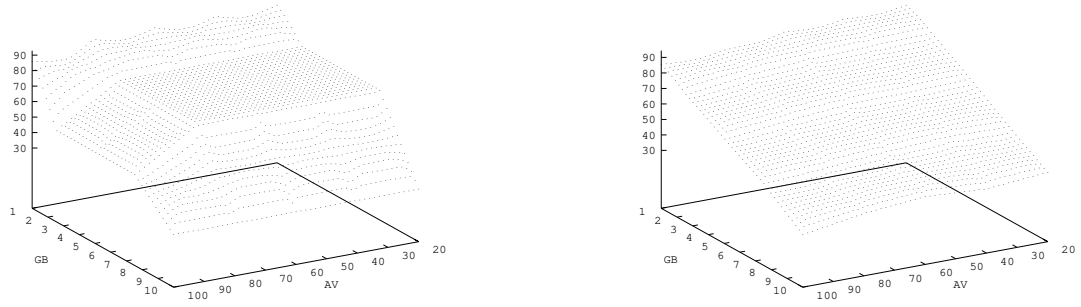
Figure 4: Inference process in FASY.



a) BTS b)



a) OTA b)



a) Folded Cascode OTA b)

Figure 5: Decision surfaces: a) Obtained from a human expert. b) Obtained from the optimization process.

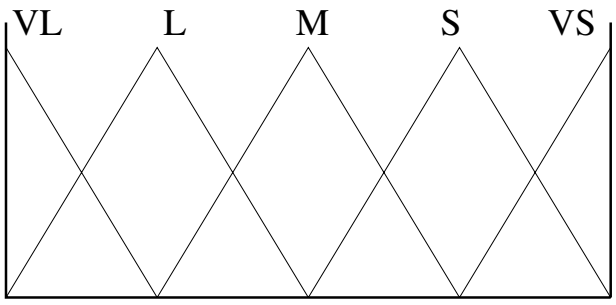


Figure 6: Membership functions used to obtain figure 5.

a)

b)

Figure 7: Superimposed decision surfaces (op.amps). a) Specified active area $< 5000\mu m^2$. b) Specified area $< 10000\mu m^2$.

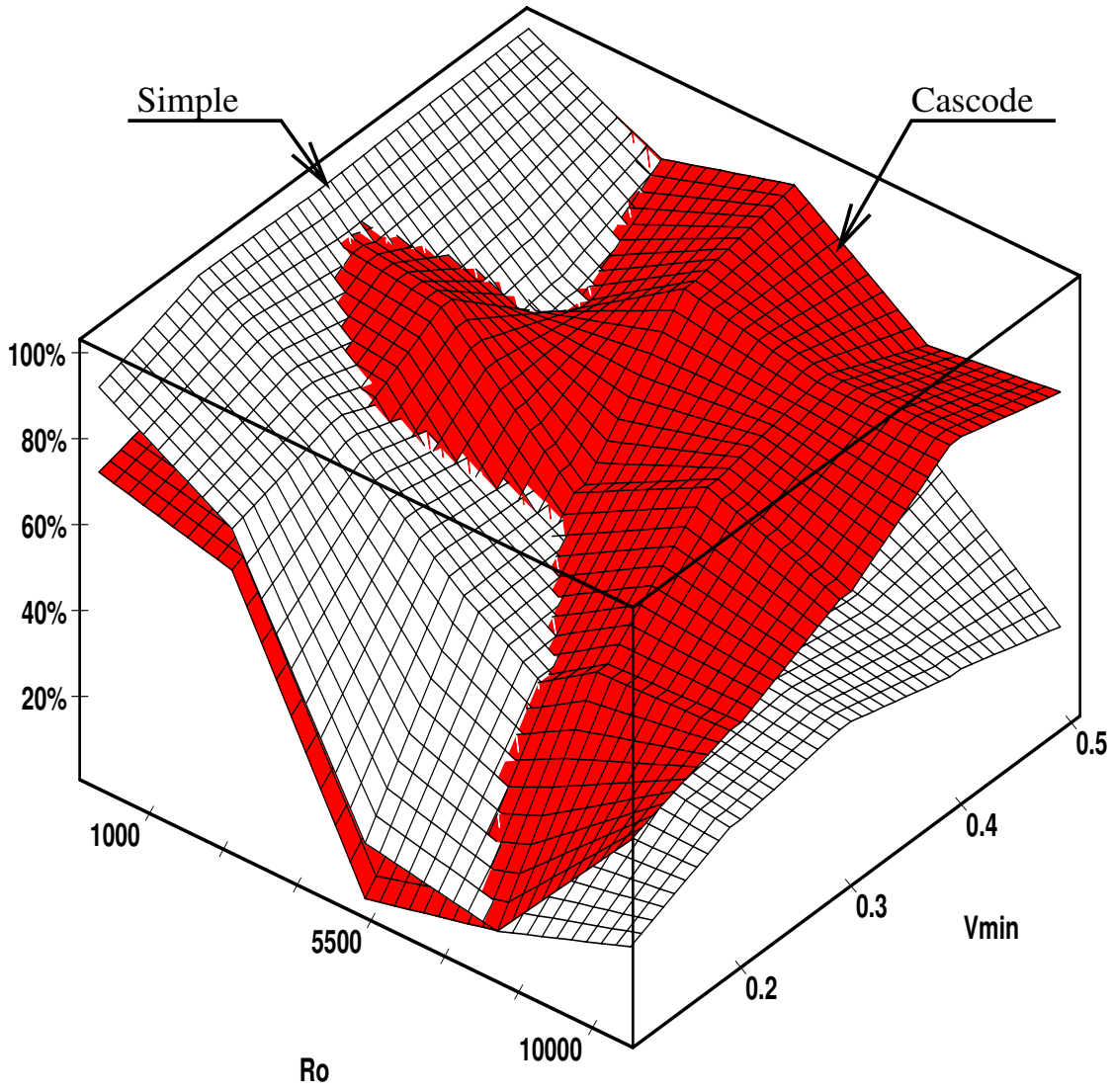


Figure 8: Superimposed decision surfaces (current mirrors). (Active area $< 5000\ \mu m^2$, output current $I_o = 10\mu A$).

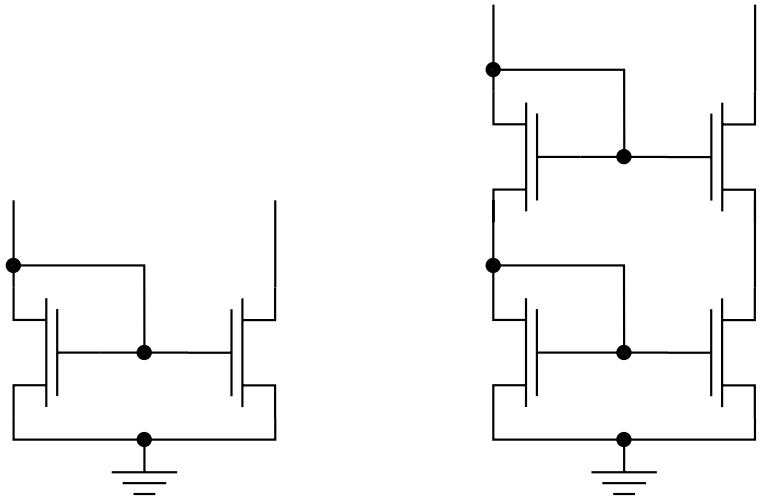
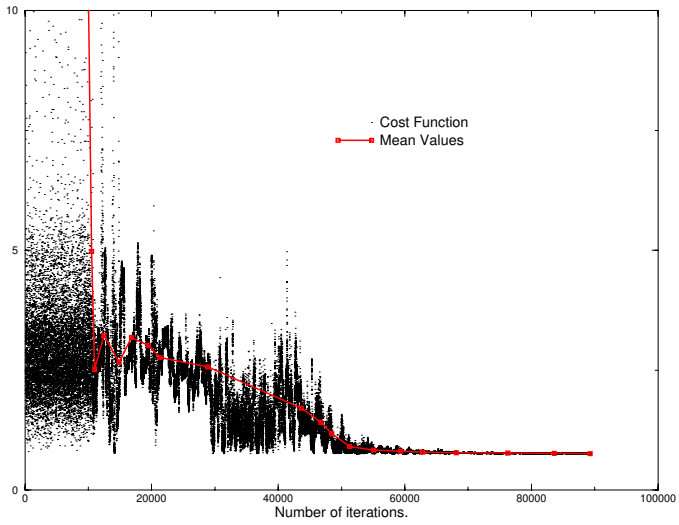
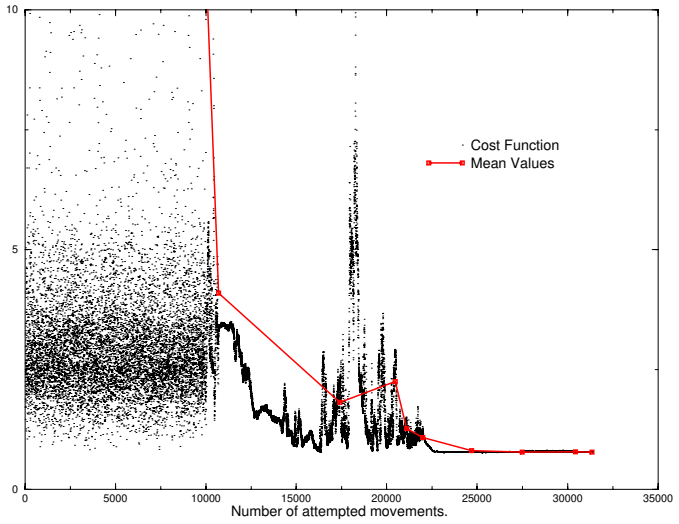


Figure 9: Basic CMOS current mirrors



a)



b)

Figure 10: Evolution of the objective function: a) Simulated annealing, b) Proposed method.

Specification name	Spec.
DC Gain (dB)	90
Unity Gain Freq. (MHz)	1
Phase Margin (deg)	60
Slew Rate (V/μsec)	1
$1/f$ Noise at 1 KHz ($nV/\sqrt{Hz}$)	150
Power Dissip. (mW)	1
Active Area (μm^2)	min.
Load Capacitance (pF)	10

Table I. An example of user specifications for opamps.

Gain / Freq.	S	M	L
S	VL	VL	M
M	VL	L	S
L	L	M	VS

a)

Gain / Freq.	S	M	L
S	VL	L	S
M	L	M	VS
L	M	S	VS

b)

Gain / Freq.	S	M	L
S	VL	L	S
M	VL	L	S
L	VL	M	S

c)

VL	*Very Large*
L	*Large*
M	*Medium*
S	*Small*
VS	*Very Small*

Table II. Decision rules defined by an expert. The tables show the adequacy of each topology in the case of a large active area. a) BTS, b) OTA, c) Folded Cascode OTA.

Specification name	Spec.	Simple	Cascode	Spec.	Simple	Cascode
Output resistance (Ohm)	1000	4990	22260	10000	22755	24617
Output voltage, V_{MIN} (V)	0.1	0.03	0.04	0.5	0.01	0.04
Output Current (μA)	10	9.8	9.4	10	2.0	8.1
Active Area (μm^2)	5000	2596	3280	5000	2816	2176
Topology selection grade	-	91	72	-	22	76

Table III. Some results obtained with FASY (current mirrors).

Specification name	Spec.	BTS	OTA	Casc	Spec.	BTS	OTA	Casc
DC Gain (dB)	60	106.8	65.62	138.3	130	124	75.8	151.7
Unity Gain Freq. (MHz)	3.75	4.34	4.94	3.61	5.00	7.04	7.73	6.28
Phase Margin (deg)	60	68.1	60.3	79.0	60	76.6	62.6	70.4
Active Area (μm^2)	5000	4890	4960	4870	10000	7800	9980	8990
Load Capacitance (pF)	10	10	10	10	10	10	10	10
Topology selection grade	-	68	59	65	-	65	37	71
Total time (sec)	-	152	122	154	-	175	136	118

Table IV. Some results obtained with FASY (op.amps).