A CPPLL Hierarchical Optimization Methodology Considering Jitter, Power and Locking Time

Jun Zou Daniel Mueller Helmut Graeb Ulf Schlichtmann
Institute for Electronic Design Automation, Techn. Univ. Muenchen, Arcisstr. 21, 80333 Munich, Germany

Abstract

In this paper, a hierarchical optimization methodology for charge pump phase-locked loops (CPPLLs) is proposed. It has the following features: 1) A comprehensive and efficient behavioral modeling of the PLL enables fast simulations and includes the important PLL performances jitter, power and locking time, as well as stability constraints for the nonlinear locking process and the linear lock-in state; 2) Behavioral modeling of the PLL building blocks addresses as behavioral-level parameters: current and jitter of the charge pump (CP), gain, current and jitter of the voltage controlled oscillator (VCO), as well as R, C´s of the loop filter (LF). It enables a proper propagation of PLL specifications down to the circuit-level design parameters; 3) An accurate and efficient performance space exploration technique on circuit level provides the feasible regions of the behavioral-level parameters of the building block by multidimensional Pareto-optimal fronts. This enables a first-time-successful top-down optimization process. Experimental results show the efficacy and efficiency of the presented method. The methodology can be applied to other large-scale analog circuits.

Categories and Subject Descriptors: B.7 [Integrated Circuits]: Design Aids; I.6.5 [Simulation and Modeling]: Model Development
General Terms: Performance, Design
Keywords: Hierarchical Optimization, Pareto-Optimal Fronts

1. Introduction

Research on automatic sizing has achieved sustained success [7-12]. With the help of deterministic or statistical optimization algorithms, a set of design parameters, e.g. transistor lengths and widths, that satisfies the performance specifications, e.g. gain, delay, can be automatically determined. During the automatic sizing process, the performances have to be evaluated for a large number of different circuit parameter sets. This performance evaluation is usually done by circuit-level simulation in order to accurately capture technological effects. For a PLL, a single transient circuit-level simulation requires 1.5 hours, hence an optimization process with 1000 simulations would last 62 days. Therefore, for large-scale analog circuits, optimization based only on circuit-level simulation is not feasible.

In order to shorten the simulation time, behavioral modeling using a hardware description language (HDL) [20] has been developed for large-scale analog circuit design. HDL descriptions of large-scale analog circuits like PLLs are frequently used for performance verification. Optimization approaches that aim at behavioral-level design parameters have been published [4, 5]. But it is commonly accepted, that an automatic optimization process should establish a complete top-down design process from the behavioral level down to the circuit level by a mixed behavioral-/circuit-level modeling [1, 2].

The challenge in such a hierarchical optimization process lies in a suitable behavioral-level modeling that captures the major effects of technology on performance. Additionally the PLL specifications must be transformed into realistic behavioral- level parameter values for the PLL blocks, so that they do not overburden the circuit-level optimization process. It is very important to include the intrinsic trade-offs in analog design, like the RF design hexagon between circuit performances [3], illustrated in Fig. 1(a), or between building blocks of a circuit, illustrated for the power distribution in a PLL in Fig. 1(b).

In this context, methods for performance space exploration (PSE) come into play [14-18]. PSE transforms technological constraints bottom-up into feasible performance values e.g. of the PLL building blocks. These feasible performance values represent the performance trade-offs and are given as Pareto-optimal fronts. While [14-18] focus on how to efficiently and accurately perform PSE by means of statistical or deterministic, simulation-based or equation-based methods, this paper presents how to apply PSE within hierarchical optimization.

An efficient analog hierarchical design flow is presented. We use PSE to calculate the Pareto-optimal fronts of each circuit building block and embed these fronts into the behavioral models of the building blocks. By an automatic sizing on behavioral level, the specifications for the circuit level are determined. Each building block is then sized separately by automatic sizing on circuit level.

Compared to the recent work in [21], which is based on a posynomial performance function modeling, the method presented here is simulation-based. While in [21], behavioral-level sizing includes the behavioral-level parameter feasibility in form of safety margin, we will include the complete Pareto-optimal fronts. Compared to the recent works in [6, 19] which consider two PLL performances and a simple trade-off modeling, in this paper a more comprehensive PLL behavioral modeling and a more comprehensive trade-off modeling up to 3-D Pareto fronts is presented.

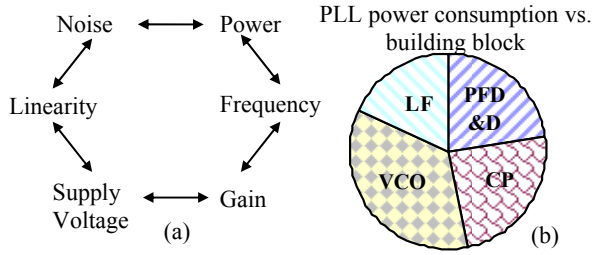

Fig. 1 Trade-offs among a) performances b) PLL blocks

DAC 2006, July 24-28, 2006, San Francisco, California, USA.

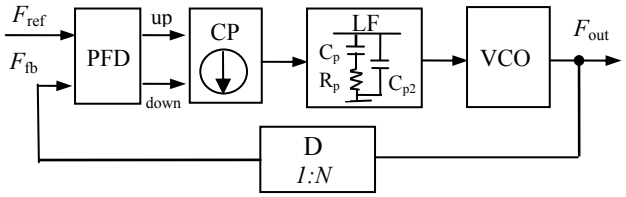

Fig. 2 Block diagram of a charge-pump PLL system

The paper is organized as follows. In Sec. 2, the CPPLL modeling for hierarchical optimization is presented. Sec. 3 describes the proposed hierarchical optimization methodology. Experimental results are given in Sec. 4. Sec. 5 concludes.

2. Charge pump PLL hierarchical modeling

2.1. Basics

PLLs play important roles in many applications ranging from frequency synthesis to clock recovery in wireless receivers. The CPPLL architecture is considered as a simple and effective design platform with advantages such as zero phase error and an extended frequency range of operation, and is widely adopted in many PLL systems.

A CPPLL consisting of five building blocks, namely phase frequency detector (PFD), charge pump (CP), loop filter (LF), voltage controlled oscillator (VCO) and divider (D) is shown in Fig. 2. The output frequency can be set to multiples of the reference input frequency by changing the ratio N of the divider: $F_{out} = N \cdot F_{ref}$.

The building block implementations are taken from [19]. The analog blocks CP, LF and VCO are considered in the design while the digital blocks (PFD&D) are assumed as ideal. The CP consists of two current sources: source and sink currents. The current steering configuration schematic of the CP is shown in Fig. 3(a). When the up (down) signal is active, the source current flows into (out of) the loop filter, so that the output voltage of the loop filter rises up (drops down), which forces a higher (lower) oscillation frequency. Note that the up and down signals cannot be active at the same time.

The VCO is a five-stage single-end ring oscillator shown in Fig. 3(b). The input voltage controls the current through the delay elements, thus determines the delay time of each stage and in turn determines the output oscillation frequency. An ideal VCO generates a periodic signal whose frequency is a linear function of the controlling voltage. The output frequency f_{out} is given by:

$$f_{out} = f_{min} + K_{VCO} \cdot (V_{in} - V_{min}) \qquad (1)$$

f_{min} is the minimum output frequency at the corresponding minimum input voltages V_{min}. V_{in} is the output controlling voltage of the loop filter.

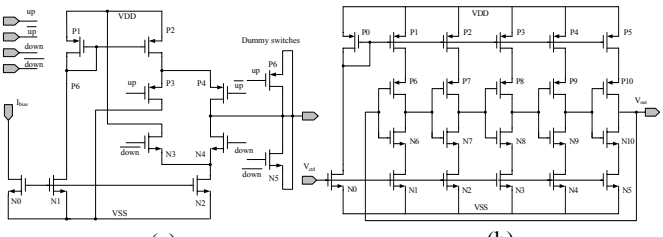

Fig. 3 (a) Charge pump schematic (b) VCO schematic

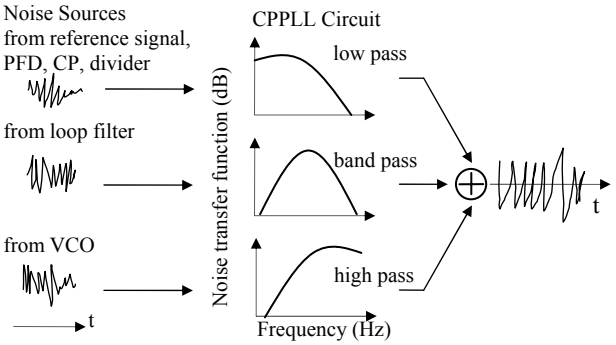

Fig. 4 Noise transfer functions of a PLL

The CPPLL is realized in a 180 nm technology with a supply voltage V_{DD} of 3V, using a reference signal f_{in} of 25MHz to generate the output frequency ranging from f_{min} (=150MHz) to f_{max} (=500 MHz). In the following we will present a hierarchical modeling, where system performances are modeled in dependence of behavioral-level parameters from each building block. These behavioral-level parameters can be determined by simulation of the blocks on circuit level. Fig. 5 gives an overview of the hierarchical performance modeling, which will be detailed in the following.

2.2. CPPLL system on behavioral level

2.2.1. System performances

In this work, we consider six important performances of the CPPLL system: *locking time*, *jitter, power consumption, unity-gain-bandwidth, phase margin* and *output frequency*.

The locking time is defined as the time taken by the CPPLL to synchronize with or to lock onto a new frequency. The performances shall be considered at the worst case. Therefore the locking time is defined as the time for the output frequency directly jumping from f_{min} to f_{max}.

Jitter is another critical performance of the PLL. Any jitter (time-domain) or phase noise (phase-domain), respectively, at the output of the PLL generally degrades the performance margins. The total CPPLL noise comes from many sources, such as from the reference signal, the CP, or the VCO. The CPPLL exhibits different noise transfer functions with respect to the different noise sources [5]. As illustrated in Fig. 4, the CPPLL acts as a high-pass filter regarding the noise from the VCO and as a low-pass filter regarding noise from the CP. Hence we have to trade-off jitter of the VCO vs. jitter of the CP when minimizing the total CPPLL jitter. In order to obtain the worst-case, we calculate the jitter of the VCO at the minimal output frequency f_{min}.

With the digital blocks PFD and D predetermined, the power consumption is defined as the sum of the power consumption in the CP, LF and VCO (we consider only analog blocks):

$$\begin{aligned} &\text{Power} = V_{DD} \cdot 4 \cdot I_{CP} + I_{CP}^2 \cdot |Z_{LF}| + V_{DD} \cdot I_{VCO} \\ &\text{where } Z_{LF} = \left(R_P + 1/s \cdot C_{P1}\right) \| 1/s \cdot C_{P2} \end{aligned}, \qquad (2)$$

where Z_{LF} is the impedance of the LF. We consider the worst- case power consumption that occurs when the CPPLL works at its maximal output frequency f_{max}.

For the optimization process we select locking time, jitter and power as optimization targets $\mathbf{f} = [f_1, f_2, \ldots, f_n]^T$ on behavioral level. The output frequency range of the PLL is determined by the VCO, therefore this performance is directly forwarded as a circuit-level performance of the VCO block.

Additionally, the following optimization constraints $\mathbf{c}_S(\mathbf{p}) \geq 0$ concerning stability are formulated:

- The stability limit of the PLL nonlinear locking process is nominally constant for various loop parameters [23] and is primarily defined by the ratio *RUR* between reference signal frequency and the unity-gain-bandwidth (UGB). We specify $RUR \geq 20$.
- In linear lock-in state, the phase margin (*PM*) of the PLL is taken as stability criterion. We specify $PM \geq 45°$.

2.2.2. Behavioral-level parameters

For each building block α=A, B, … of the CPPLL, we define a set of system parameters $\mathbf{p}_\alpha$ that is capable of capturing the influence of the blocks on the system performances & tradeoffs:

- CP: outside-biased current I_{CP}, jitter J_{CP}
- VCO: gain K_{VCO}, current consumption I_{VCO}, jitter J_{VCO}
- LF: filter elements R_P, C_P, C_{P2}
- D: divider value N

The collection of the parameters of all blocks is denoted by $\mathbf{p} = [\mathbf{p}_A^T\ \mathbf{p}_B^T \cdots \mathbf{p}_Z^T]^T$.

2.2.3 Behavioral-level simulation

The behavioral models of the CPPLL building blocks in [19] and injecting noise within transient simulation [13] are used for an efficient simulation of the CPPLL on behavioral level. The behavioral-level simulation provides a mapping of the behavioral-level parameters onto the system performances: $\mathbf{p} \rightarrow \mathbf{f}$.

2.3. CPPLL building blocks on circuit level

On circuit level, the behavioral-level parameters become performances of the building blocks that can be simulated in dependence of circuit-level design parameters. For the CP for instance, the absolute values of the simulated charging and discharging currents I_{up} and I_{down} are equal to the value of the outside-biased current I_{CP} on behavioral level, and the behavioral-level jitter J_{CP} becomes a circuit-level performance of the CP. For the VCO, gain, current consumption, jitter and output frequency range are the circuit-level performances corresponding to the behavioral-level parameters.

The transistor widths/lengths are the circuit-level parameters. Circuit simulation provides a mapping of the transistor widths and lengths $\mathbf{x}_\alpha$ onto the behavioral-level parameters $\mathbf{p}_\alpha$ of each block α: $\mathbf{x}_\alpha \rightarrow \mathbf{p}_\alpha$, α=*A, B, …, Z.*

3. Hierarchical optimization methodology

For large-scale analog circuits, that are modeled mixed on behavioral and circuit level, two optimization strategies have been published. The first strategy treats the parameters on behavioral and circuit level at the same time [21, 26]. The second performs a strict two-stage process: In step one behavioral-level parameters are optimized using behavioral-level simulation. In step two, by taking the resulting optimal behavioral-level parameter values as specifications for the circuit-level performances of each block, circuit-level parameters of each block are optimized separately [6, 15]. Which of these two strategies is better, certainly depends on the targeted application. This paper follows the second strategy for the following reasons:

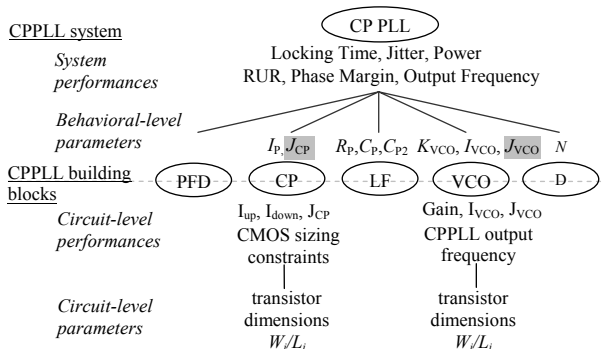

Fig. 5 Hierarchical performance modeling of the CPPLL

- Re-use of building blocks (also in system classes other than PLLs) may be more easy.
- The clear distinction between system requirements and building-block requirements enables a deeper insight into the complex trade-offs in an interactive design process.
- Different building-block implementations can easily be investigated.

In the two-stage optimization strategy, a new task arises which consists in preventing the optimization from producing unrealistic requirements on the blocks. To solve that problem, we propose a preprocessing phase "*bottom-up extraction*" before the two-stage *top-down sizing*. Fig. 6 illustrates the bottom-up extraction phase and the two-stage top-down sizing process, described in detail in the following.

3.1. Bottom-up extraction on circuit level

Based on test benches for the circuit-level simulation of each building block[1] and on performance space exploration (PSE), e.g. [14], the Pareto-optimal fronts for the performances of the individual building blocks are calculated. Essentially, PSE transforms technological constraints bottom-up into feasible performance values of the building blocks, denoted as:

$$\left\{\mathbf{x}_\alpha \middle| \mathbf{c}_{\mathbf{x}\alpha}(\mathbf{x}_\alpha) \geq 0\right\} \mapsto \left\{\mathbf{p}_\alpha \middle| \mathbf{c}_{\mathbf{p}\alpha}(\mathbf{p}_\alpha) \geq 0\right\},\ \alpha=A,B,C,\cdots,Z \quad (3)$$

where $\mathbf{c}_{\mathbf{x}\alpha}(\mathbf{x}_\alpha)$ determines the sizing constraints on circuit level, e.g. the feasible ranges for transistor width/length or DC conditions for transistor in saturation region, and where $\mathbf{c}_{\mathbf{p}\alpha}(\mathbf{p}_\alpha) \geq 0$ determines the feasible regions for $\mathbf{p}_\alpha$.

The Pareto-optimal front is the part of the boundary of the feasible region that describes the set of optimal trade-offs between competing objectives. A Pareto-optimal point is a combination of block performance values, where it is not possible to improve one performance without deteriorating another one.

3.2. Bottom-up extraction on behavioral level

From [22] it follows that an optimal behavioral-level parameter value will always be a Pareto-optimal point of a building block. Accordingly, an efficient setup for the optimization process at behavioral level restricts the behavioral-level parameters $\mathbf{p}$ to walk along the Pareto-optimal front during optimization. Hence, our behavioral models consist of two parts:

- the basic function of the building block, and
- the extracted Pareto-optimal front, represented by a look-up table or a mathematical expression.

[1] As shown in Fig. 6, the test benches are also required for the circuit simulation that is used in circuit-level optimization.

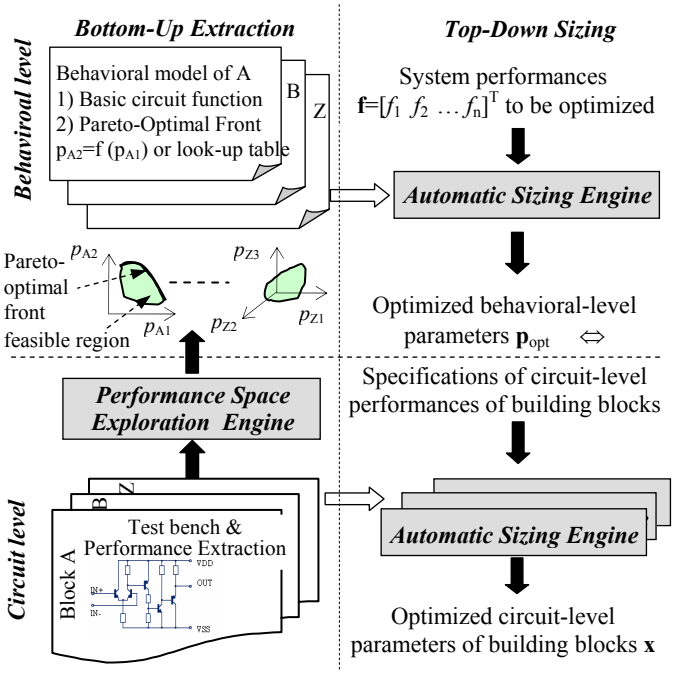

Fig. 6 Hierarchical optimization methodology

In this way, the number of behavioral-level parameters is reduced by one for each building block with Pareto-optimal fronts, which contributes to a higher efficiency of the optimization process. The behavioral models are used for the system simulation within the behavioral-level optimization.

3.3. Top-down sizing on behavioral level

System performances are optimized subject to behavioral-level constraints by sizing of the behavioral-level parameters **p**:

$$\max_{\mathbf{p}} \mathbf{f}(\mathbf{p}) \text{ s.t. } \mathbf{c}(\mathbf{p}) \geq 0$$

$$\text{with } \mathbf{c}(\mathbf{p}) = \left\{ \mathbf{p} \middle| \mathbf{c}_S(\mathbf{p}) \geq 0 \wedge \mathbf{c}_{\mathbf{p}\alpha}(\mathbf{p}_\alpha) \geq 0, \alpha = A, B, \ldots, Z \right\} \quad (4)$$

$$\mapsto \mathbf{p}_{\text{opt}},\ \mathbf{f}_{\text{opt}} = \mathbf{f}(\mathbf{p}_{\text{opt}})$$

The behavioral-level constraints include both, sizing constraints from the circuit level that were extracted by PSE, and optimization constraints for system performances (as formulated for PLL stability). For optimization, a standard optimization tool is used [12]. The obtained optimal behavioral-level parameter values $\mathbf{p}_{\text{opt}}$ are then propagated to performance specifications on circuit level.

3.4. Top-down sizing on circuit level

Performance specifications propagated from the behavioral-level sizing process are achieved by simultaneous circuit-level optimization of all building blocks, each considering the circuit-level technological constraints:

$$\min_{\mathbf{x}\alpha} \left\| \mathbf{p}(\mathbf{x}_\alpha) - \mathbf{p}_{\text{opt}} \right\| \text{ s.t. } \mathbf{c}_{\mathbf{x}\alpha}(\mathbf{x}_\alpha) \geq 0,\ \alpha = A, B, \ldots, Z$$

$$\mapsto \mathbf{x}_{\text{opt}},\ \mathbf{p}_{\text{opt}} = \mathbf{p}(\mathbf{x}_{\text{opt}}) \quad (5)$$

Having started with a system specification, we end up with a fully sized circuit implementation on circuit level.

3.5. Bottom-up design validation

Finally, the sized CPPLL should be validated. A circuit-level simulation would be most accurate but is prohibitively expensive. Alternatively we propose to calibrate the behavioral models. After extracting the actual performance values of each building block, the behavioral models are updated by directly using these values instead of the models of the Pareto-optimal front. Then, the final system performance can be extracted from a behavioral-level simulation based on the calibrated models.

In the following, experimental results for a CPPLL design are presented that demonstrate the efficacy and efficiency of the presented hierarchical optimization methodology.

4. Experimental results

4.1. Behavioral-level simulation

Behavioral modeling offers the basis for a fast optimization process. For the locking time, 20µs transient system time have to be simulated. The behavioral modeling from [19] results in 40 seconds simulation time, compared to about 90 minutes for circuit-level simulation.

The behavioral jitter modeling based on [13] results in some minutes simulation time, compared to some days for circuit-level simulation.

4.2. Pareto-optimal fronts in the behavioral models

According to the behavioral-level parameter set described in Sec. 2.2.2 (Fig. 5), we are facing a 2-D Pareto front for the CP and a 3-D Pareto front for the VCO. Based on PSE according to [14], discrete points of the Pareto fronts are calculated. Then, by means of curve fitting, analytical formulations of the Pareto fronts are determined. For 3-D, we apply TableCurve 3D [26]. The simulation time to compute the Pareto fronts was 5-6 hours.

Fig. 7 shows the Pareto front for the competing objectives jitter and current of the CP. As both jitter and current should be a small as possible, the given Pareto front represents the lower bounds of combined jitter and current values that can be achieved with the given CP implementation.

Fig. 8 shows the feasible ranges bounded by the Pareto fronts for the competing objectives jitter, gain and current of the VCO in 3-D and in the different 2-D projections. Please note that it is a non-trivial task to compute 3-D Pareto fronts. The results here are based on circuit-level simulation and represent accurate values. We can evaluate e.g. the decreasing trend of J_{VCO} with increasing I_{VCO} from Fig. 8(c), or, how K_{VCO} increases at the cost of a larger J_{VCO} for constant I_{VCO} from Fig. 8(d).

The data in Fig. 7 and Fig. 8 provides valuable insight into the design trade-offs to the designer and can be reused for other applications than PLLs.

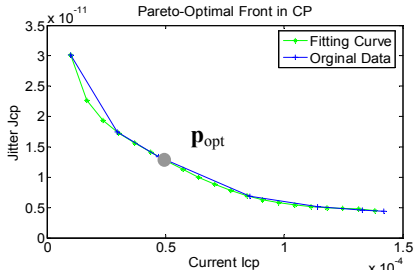

Fig. 7 Jitter and current trade-off in charge pump

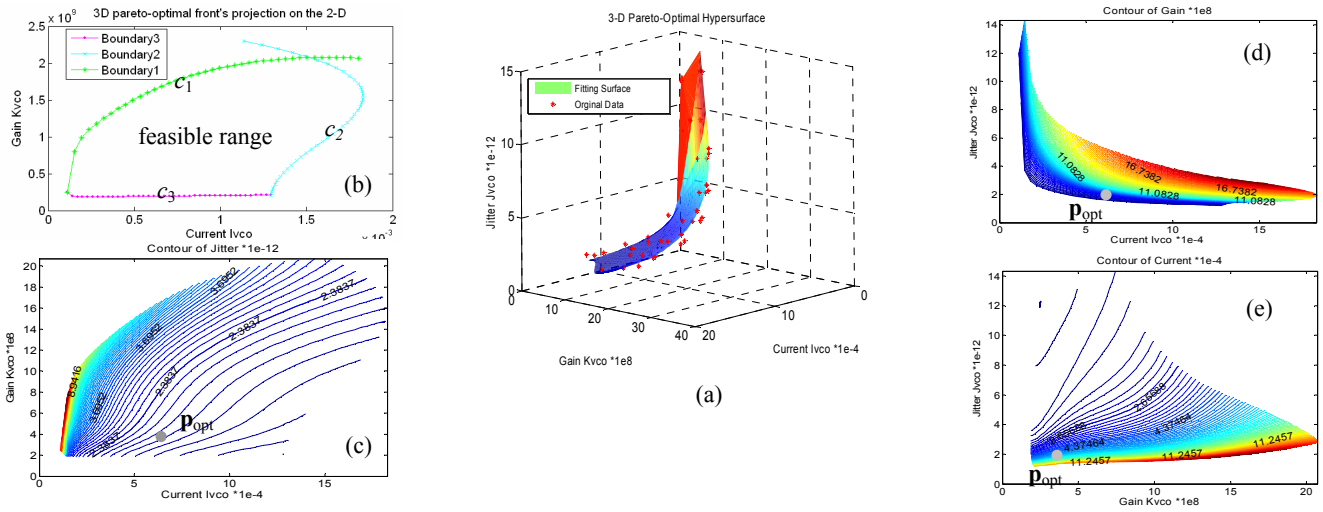

Fig. 8

Pareto-optimal Front of jitter, gain and current of VCO (a) 3-D (b) projection on gain vs. current (c) projection on gain vs. current plus contour lines of jitter (d) projection on jitter vs. current plus contours of gain (e) projection on jitter vs. gain plus contours of current

As mentioned in Sec.3.2, we can assume that the optimal solution will be on the Pareto front. Therefore, we calculate explicit analytical models for the Pareto fronts of the behavioral-level parameters and thus reduce the design space. Here we express the parameters J_{CP} and J_{VCO} (marked in gray in Fig. 5) in dependence of the other parameters of the respective block. The resulting expressions become a part of the behavioral block models given in List 1 and List 2, which in this way include the Pareto fronts extracted on circuit level. The independent behavioral-level parameter set now is $\mathbf{p} = [I_{CP}, K_{VCO}, I_{VCO}, R_P, C_P, C_{P2}, N]$.

List 1 Behavioral model of CP in Verilog-A

```
module CP(Iout, Down, N_Down, N_Up, Up, Ibias);
    ……
    parameter real Icp=25.0e-6;      //charge pump's output current
    parameter real  a= 9.740357938125466 ; // curve fitting coefficients
    .......
    parameter real  d= 72.72517183514785 ;
    integer state;       // charge pump state: "-1->charge","1->discharge",
    real Jcp, It;
     ……
analog begin
    @(cross(V(Up)-VTH, 1)) begin        state = -1; end
    @(cross(V(Down)-VTH, 1)) begin     state = 1;  end
    @(cross(V(Up)-VTH, -1)) begin       state =0;   end
    @(cross(V(Down)-VTH, -1)) begin    state =0;   end
     ……
// define Jcp as a polynomial function of current Icp
It  = Icp*1e6;
    Jcp = (a + b* It + c*pow(It,1.5) + d*pow(It,0.5)) *1e-12;
    dt=Jcp * $dist_normal(seed,0,1);
……
    I(Iout)<+transition(Ip*state*(1+state*Mis), Delay+Jcp, TransTime );
    ……
end
endmodule
```

List 2 Behavioral model of VCO in Verilog-A

```
module VCO(in,out);
    ……
    parameter real Kvco=600e6;
    parameter real Ivco=100e-6;
    parameter real  a= 65.67689290353322; // fitting curve coefficients
    ……
    parameter real  i= -3.982846605714493;
    real freq, phase, dT, delta, Vout , Fmax, TKvco, Jvco, It;
analog begin
    ……
freq=(V(in)-Vmin)* Kvco + Fmin;
    .....
// VCO's Jitter as a function of Ivco, Kvco
TKvco=Kvco/1e8;    It=Ivco*1e4;
Jvco=a+(b/sqrt(It))+(c/pow(It,1.5))+(d/pow(It,2))+(e*TKvco)+(f*TKvco*ln(TKvco)
)+(g*exp(TKvco/(w)))+(h*sqrt(TKvco)*ln(TKvco))+(i*ln(TKvco)*ln(TKvco));
Jvco= (1/Jvco) * 1e-12;
delta=Jvco*sqrt(2*N);
freq= (freq/N) / (1 + dT * freq / N);
phase=idtmod(freq, 0.0, 1.0, -0.5);
@(cross(phase-0.25,1,ttol))begin
                dT=delta*$dist_normal(seed,0,1);
                Vout=VDD;     end
    @(cross(phase+0.25,1,ttol))begin
                dT=delta*$dist_normal(seed,0,1);
                Vout=VSS;     end
    V(out)<+transition(Vout,0,tt);
end
endmodule
```

4.3. Hierarchical optimization

Combining these behavioral models, transient simulation with Spectre is used to obtain the locking time T_s, while another transient simulation is for the jitter J_{SUM}. Based on *s-domain* analysis of the PLL, the linear lock-in state properties, e.g. unity-gain-bandwidth, phase margin, are obtained. A third simulation with *octave* [24] is used for the behavioral-level constraints: the ratio *RUR*, the phase margin *PM* and the feasible range boundaries, which consist of c_1, c_2 and c_3 in Fig. 8(a). The optimization process is realized in *WiCkeD* [12]. The behavioral-level optimization and circuit-level optimization each requires about 1.5 hours, which results in a total of 3 hours.

Table 1 shows the results of the two optimization phases. Starting from the system specifications in column 3 and the initial design in column 4, behavioral-level optimization leads to the optimized system performances in the upper half of column 5 and the behavioral-level parameters in the lower half of column 5. The latter, propagated as specifications to the circuit-level optimization, results in the final $\mathbf{P}_{opt}$, which are listed in the lower half of the last column and depicted in dots in Fig. 7 and Fig. 8, respectively. A system simulation with the behavioral models calibrated according to the circuit-level design parameters yields the system performances in the upper half of column 6. We can

see that by considering the technological feasibility of building blocks a first-time-successful hierarchical optimization has been achieved for a comprehensive set of design aspects.

5. Conclusion

We have presented a comprehensive hierarchical modeling of a PLL on behavioral and circuit level. As a key feature we are considering more important performances, i.e. jitter, current, locking time and stability, and the building block contributions to them.

A hierarchical optimization methodology has been described and successfully applied to the sizing of a PLL. As key features, we have applied advanced performance space exploration techniques to calculate the Pareto-optimal fronts in 2-D and 3-D and we have integrated the Pareto-front models in the hierarchical optimization process.

The computational costs of the optimization process are within a few hours and thus very practicable.

The presented method is applicable to other large-scale analog circuits and helps shortening the optimization process while considering more performance features.

References

[1] H. Chang, A. Sangiovanni-Vincentelli, F. Balarin, etc, "A Top-down, Constraint-Driven Design Methodology for Analog Integrated Circuits", *IEEE CICC*, pp. 533-536, 1992.

[2] G. G. E. Gielen and R. A. Rutenbar, "Computer-Aided Design of Analog and Mixed-Signal Integrated Circuit", *Proceeding of The IEEE,* VOL. 88 NO.12 Dec. 2000.

[3] B. Razavi, *RF Microelectronics,* McGRAW-Hill, 2003.

[4] M. H. Perrott, "Fast and Accurate Behavioral Simulation of Fractional-N Synthesizers and other PLL/DLL Circuit", *IEEE DAC*, pp. 498-503, June 2002.

[5] M. Mansuri and C. K. K. Yang, "Jitter Optimization Based on Phase-Locked Loop Design Parameters", *IEEE Journal of Solid-State Circuits*, VOL.37, No.11, Nov. 2002.

[6] S. K. Tiwary, S.Velu, R. A. Rutenbar and T. Mukherjee, "Pareto Optimal Modeling for Efficient PLL Optimization", *Nanotech 2004,* Vol. 2 pp. 195 – 198.

[7] W. Nye, D. Riley, and A. Sangiovanni-Vincentelli, "An optimization-based system for the design of integrated circuits", *ICCAD,* 1988.

[8] K. Antreich, J. Eckmueller, H. Graeb, M. Pronath, etc, "WiCkeD: Analog circuit synthesis incorporating mismatch", *IEEE CICC*, pp. 511-514, May 2000.

[9] M. d. M. Hershenson, S. P. Boyd and T. H. Lee, "Optimal Design of a CMOS Op-Amp Via Geometric Programming", *IEEE Transactions on Computer-Aided Design of Integrated and System,* Vol.20, No.1, January 2001.

[10] R. Phelps, M. Krasnicki, R.A. Rutenbar, and L.R. Carley, and J.R. Hellums; "Anaconda: simulation-based synthesis of analog circuits via stochastic pattern search", *IEEE Trans. CAD,* 2000.

[11] G. Van der Plas, G. Debyser, F. Leyn, K. Lampaert, J. Vandenbussche, G. Gielen, W. Sansen, P. Veselinovic, D. Leenaerts; "AMGIE-A synthesis environment for CMOS analog integrated circuits", *IEEE Trans. CAD*, 2001.

[12] *WiCkeD,* Available from *www.muneda.com*

[13] K. Kundert, "Predicting the phase noise and jitter of PLL-based frequency synthesizers", *http://www.designers-guide.org/*

[14] D. Müller, G. Stehr, H. Graeb, U. Schlichtmann, "Deterministic Approaches to Analog Performance Space Exploration", *IEEE DAC,* 2005.

[15] G. Stehr, H. Graeb and K. Antreich, "Performance trade-off analysis of analog circuits by Fourier-Motzkin elimination with application to hierarchical sizing", *IEEE ICCAD,* pp.847-854, 2004.

[16] B. D. Smedt, G. E. Gielen, "WATSON: Design Space Boundary Exploration and Model Generation for Analog and RF IC Design", *IEEE Trans. CAD*, 2003.

[17] F. De Bernardinis, A. Sangiovanni-Vincentelli, "A Methodology for System-Level Analog Design Space Exploration", *DATE,* 2004.

[18] M. Hershenson, "Efficient description of the design space of analog circuits," *IEEE/ACM DAC*, pp.970-973, 2003.

[19] 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", *IEEE Behavioral Modeling and Simulation (BMAS)*, 2005.

[20] *Verilog-AMS Language Reference Manual: Analog & Mixed-Signal Extensions to Verilog-HDL,* version 2.1. Available from *www.verilog-ams.com.*

[21] X. Li, J. Wang, L.T. Pileggi, T.S. Chen and W.J. Chiang, "Performance-Centering Optimization for System-Level Analog Design Exploration", *IEEE ICCAD,* 2005.

[22] T. Eeckelaert, T. McConaghy, G. Gielen, "Efficient Multiobjective Synthesis of Analog Circuits using Hierarchical Pareto-optimal Performance Hypersurfaces", *IEEE DATE,* 2005.

[23] P. K. Hanumolu, M. Brownlee, K. Mayaram and U.K. Moon, "Analysis of Charge-Pump Phase-Locked Loops", *IEEE Transactions on Circuits and Systems,* Vol. 51, No. 9, Sept. 2004.

[24] *www.octave.org.*

[25] *http://www.systat.com/products/TableCurve3D/*

[26] M. Hershenson, "Design of pipeline analog-to-digital converters via geometric programming", *IEEE ICCAD,* 2002.

Table 1 Hierarchical optimization results of CPPLL

1	2	3	4	5	6
System & blocks	Performances	Specifications	Initial design	Behavioral-level optimization	Circuit-level optimization
System: CPPLL	Locking Time T_s (µs)	≤ 2.5	2.85	2.13	2.2
	Jitter J_{SUM} (ps)	≤ 2.5	20.8	2.25	2.21
	Power (mW)	≤ 2.5	3.02	2.42	2.417
	Ratio *RUR*	≥ 20	30.6 @ f_{min} 78.6 @ f_{max}	28.5 @ f_{min} 72.8 @ f_{max}	28.7 @ f_{min} 73.5 @ f_{max}
	Phase Margin *PM* (°)	≥ 45	43.6 @ f_{min} 48.0 @ f_{max}	45.1 @ f_{min} 45.1 @ f_{max}	45.1 @ f_{min} 45.0 @ f_{max}
Block: CP	Current I_{CP} (µA)		50	47.3	I_{up}=46.5 I_{down}=-46.5
	Jitter J_{CP} (ps)		55.2	13.34	14.4
Block: VCO	Gain K_{VCO} MHz/V		270.3	347.1	348.4
	Current I_{VCO}(µA)		805.9	619.2	619
	Jitter J_{VCO} (ps)		13.7	1.88	1.876
Block: LP	Resistor R_p (k)		20	16.98	16.98
	Capacitors C_p & C_{p2} (pF)		50 & 8	44.5 & 7.66	44.5 & 7.66