

# Optimization of Integrated Spiral Inductors Using Sequential Quadratic Programming \*

Yong Zhan and Sachin S. Sapatnekar  
Department of Electrical and Computer Engineering  
University of Minnesota  
200 Union Street SE  
Minneapolis, MN 55455  
{yongzhan, sachin}@ece.umn.edu

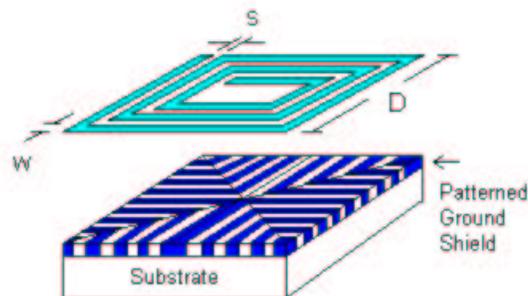
## Abstract

*The optimization of integrated spiral inductors has great practical importance. Previous optimization methods used in this field are either too slow or depend on very simplified assumptions in the device modeling which result in the algorithm only applicable to low-frequency cases. In this paper, we propose using the sequential quadratic programming (SQP) approach to optimize the on-chip spiral inductors. A physical model based on first principles is used in the back-end device-parameter extraction engine which makes the algorithm suitable to the optimization at any frequency range. In addition, compared with enumeration, which is used in many inductance optimization packages, our experiments show that the SQP algorithm can achieve at least an order of magnitude speedup while maintaining the same quality of the optimized design.*

## 1 Introduction

On-chip spiral inductors are key components in many RF integrated circuits (RFIC's) running at GHz frequency range. During the past few years, the design and optimization of integrated spiral inductors has attracted much interest in both the IC design and the electronic design automation communities. The goal of the inductor optimization may vary depending on the applications. It could be high quality factor  $Q$ , small area occupied by the device, or small parasitic effects, etc. In this paper, we concentrate on the optimization of high- $Q$  spiral inductors. Roughly speaking, there are three major loss mechanisms that degrade the quality factor of on-chip inductors: the energy loss due to the series resistance of the spiral itself, the electric coupling between the spiral and the substrate, and the magnetically induced eddy current flowing in the substrate. In [1], the substrate loss due to eddy current is significantly reduced by inserting a patterned ground shield between the spiral and the substrate as shown in Fig. 1, and

in [2], both the electric and magnetic couplings to the substrate are practically eliminated by etching away the substrate beneath the spiral. The energy loss due to the series resistance of the spiral, however, can only be reduced by optimizing the geometrical parameters of the inductor such as the number of turns, the outer length, the width of the metal traces constituting the spiral, and the space between adjacent metal traces.



**Figure 1. A three turn square spiral inductor with a patterned ground shield**

There are several previous works targeting on the optimization of the geometrical parameters of integrated spiral inductors to increase the quality factor. Enumeration is used in [3] where the geometrical parameters are first discretized, then each combination of the resulting parameter values is simulated, and finally, the parameters that result in the highest  $Q$  are used in the design. This method, although intuitive and simple, can be highly inefficient, especially when the number of adjustable parameters becomes large because the time complexity of the enumeration method is exponential with respect to the number of optimization variables. In [4], geometric programming is used to solve the spiral inductor optimization problem. Geometric programming is a powerful mathematical programming method based on the assumption that both the objective function and the constraints are posyn-

\*This work is supported in part by the NSF under award CCR-0098117

omial functions. To satisfy this requirement, the inductance of the spiral must be extracted using the approximate formula derived from curve-fitting a large number of pre-fabricated or pre-simulated designs. In addition, the entire device must be represented by a single  $\pi$  model with all the lumped parasitic components written out in posynomial functions. The limitation of this method is that several high frequency effects which are significant at GHz frequency range cannot be taken care of by the simple closed form formulas, e.g., in writing the inductance and the parasitic resistance of the spiral as posynomial functions, it is impossible to model the proximity effect which may become significant at frequencies as low as a few hundred MHz.

In this paper, we propose using SQP to optimize the quality factor of integrated spiral inductors. SQP is an iterative mathematical programming technique based on the observation that almost any smooth continuous function can be locally approximated by a quadratic function and it has the desired property that the local convergence rate is superlinear if the starting point of the iterations is close enough to the optimal solution. Compared with enumeration, the SQP algorithm achieves at least an order of magnitude speedup which can significantly reduce the turn-around time of the design of spiral inductors, and compared with the geometric programming approach, SQP can be used with any physical model to optimize the device operating at any frequency, which makes it suitable to a broader range of applications. The SQP optimizer is built upon a spiral inductor extraction engine similar to that used in [5]. The quality factor and the effective inductance of the device are extracted from the two port  $Y$  parameters, and the well-known proximity effect and skin effect are taken care of automatically by meshing the metal traces in the longitudinal direction. We assume that a patterned ground shield exists beneath the inductor and thus the eddy current in the substrate is not modeled. In addition, to make the implementation simple, we have only worked on the square spiral inductors. The rest of the paper will be organized as follows. Section 2 formulates the inductance optimization problem. Section 3 presents the computation of the quality factor, effective inductance, and sensitivity of the spiral. Section 4 introduces the SQP algorithm. Section 5 shows the experiment results, and the conclusion is given in section 6.

## 2 Problem formulation

Fig. 1 shows the schematic of a square spiral inductor. Let  $n$ ,  $D$ ,  $w$ , and  $s$  be the number of turns, outer length, width of metal traces, and the space between metal traces of the spiral, respectively. Let  $Q(n, D, w, s)$  and  $L(n, D, w, s)$  be the quality factor and the inductance of the spiral, and let  $L_{exp}$  and  $\delta$  be the targeting inductance value and the tolerance allowed for the inductance to deviate from the targeting value. Then the inductance optimization problem is formulated as

$$\begin{aligned} & \text{maximize} && Q(n, D, w, s) \\ & \text{subject to} && (1 - \delta)L_{exp} \leq L(n, D, w, s) \leq (1 + \delta)L_{exp} \end{aligned}$$

$$\begin{aligned} 2n(w + s) &\leq D & (1) \\ D_L &\leq D \leq D_U \\ w_L &\leq w \leq w_U \\ s_L &\leq s \leq s_U \end{aligned}$$

Here,  $D_L$ ,  $D_U$ ,  $w_L$ ,  $w_U$ ,  $s_L$ , and  $s_U$  are the lower and upper bounds of the corresponding optimization variables, respectively. The number of turns  $n$  is treated as a parameter rather than a variable because it can only take discrete values.  $D$ ,  $w$ , and  $s$  are treated as continuous variables for the optimization purposes and may be rounded to the nearest grid point when doing the layout. Since  $Q$  and  $L$  are both nonlinear functions of the optimization variables, our formulation gives a nonlinear optimization problem, which will be solved using the SQP algorithm. To successfully apply the SQP method, we must be able to accurately compute  $Q$ ,  $L$  and their sensitivities with respect to  $D$ ,  $w$ , and  $s$ . We will first discuss how to extract  $Q$  and  $L$ , and perform the sensitivity analysis in the next section. The detailed description of the SQP algorithm will be left to section 4.

## 3 Extraction engine

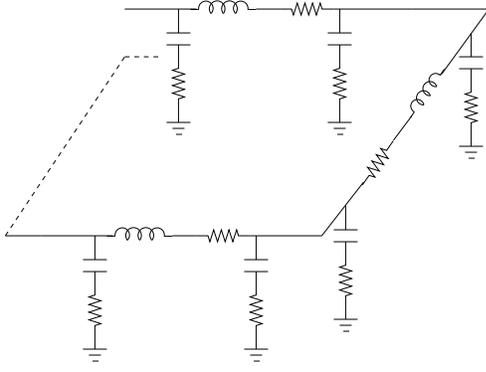
### 3.1 Inductance and quality factor extraction

For the simplicity of implementation, we have only considered the square spiral inductors in this work. The extension to other geometries such as octagonal spirals is straightforward although computationally more complicated. To extract the inductance and quality factor, the spiral is first divided into a series connection of metal segments. The length of each segment should not exceed a small fraction of the wavelength of the EM wave at the operating frequency of the device such that it is meaningful to model the segments by lumped circuit elements. Fig. 2 shows the schematic of the inductor model we used in the extraction where each segment is represented by an equivalent  $\pi$  model. The series branches include the self inductance and the parasitic resistance of the metal segments themselves, and the parallel branches include the coupling capacitance to the substrate and the conductance of the substrate. In addition, there are mutual inductance between parallel segments, and the coupling capacitance between parallel segments in adjacent turns are also modeled.

From [5], we know that using modified nodal analysis (MNA), the governing equation of the circuit can be written as

$$\begin{pmatrix} Y^C & D^T & -B^T \\ D & -Z^L & 0 \\ B & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{V} \\ \mathbf{I} \\ \mathbf{I}_V \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ V_1 \\ V_2 \end{pmatrix} \quad (2)$$

where  $Y^C$  is the AC conductance matrix of the parallel branches,  $Z^L$  is the AC impedance matrix of the series branches,  $D$  is an upper bidiagonal matrix with 1's on the diagonal entries and -1's on the superdiagonal entries and



**Figure 2. A distributed  $\pi$  model for square spiral inductors**

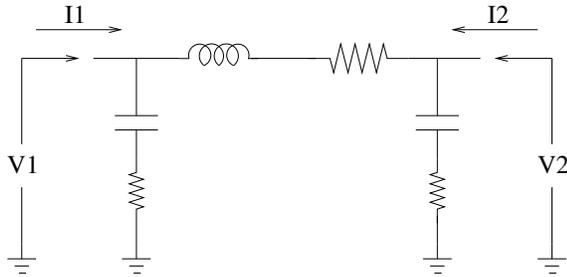
$B = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix}$ .  $\mathbf{V}$  and  $\mathbf{I}$  are the nodal voltage vector and branch current vector, respectively.  $\mathbf{I}_V$  is the vector of the current flowing through the two voltage sources  $V_1$  and  $V_2$ .

If the entire spiral is modeled by an equivalent  $\pi$  circuit as shown in Fig. 3, then the inductance and the quality factor are given by

$$L = -\frac{1}{2\pi f} \text{Im}\left(\frac{1}{Y_{12}}\right) \quad (3)$$

and

$$Q = -\frac{\text{Im}(Y_{11})}{\text{Re}(Y_{11})} \quad (4)$$



**Figure 3. Equivalent  $\pi$  model of the spiral**

The two port  $Y$  parameters can be extracted from (2). Setting  $V_1 = 1$  and  $V_2 = 0$ ,  $Y_{11}$  is given by  $Y_{11} = \left(\frac{I_{V_1}}{V_1}\right)_{V_2=0} = (I_{V_1})_{V_1=1, V_2=0}$ , while setting  $V_1 = 0$  and  $V_2 = 1$ ,  $Y_{12}$  is given by  $Y_{12} = \left(\frac{I_{V_1}}{V_2}\right)_{V_1=0} = (I_{V_1})_{V_1=0, V_2=1}$ . The total computational cost is one LU factorization of the coefficient matrix in (2) and two forward/backward substitutions. The method we used to build the  $Y^C$  and  $Z^L$  matrix is similar to that in [5] and we briefly recapitulate it here. The  $Y^C$  matrix elements are given by

$$Y_{ij}^C = \begin{cases} -\frac{1}{Z_{ij}^C} & \text{if } i \neq j \\ \sum_{k=0}^{n+1} \frac{1}{Z_{ik}^C} & \text{if } i = j \end{cases} \quad (5)$$

where  $Z_{ij}^C$  is the total impedance between node  $i$  and node  $j$  excluding that of the series branches, and  $n + 1$  is the total number of nodes excluding the ground node. All the distributed parasitic capacitances are calculated using the simple parallel plate capacitor equation. More accurate expressions can be found in [6] or use a capacitance extraction package such as FastCap [7]. To construct the  $Z^L$  matrix, we divide the relative position of metal segment  $i$  and  $j$  into three categories.

Category 1: Segment  $i$  and  $j$  are perpendicular to each other.

$$Z_{ij}^L = 0 \quad (6)$$

because there is no mutual inductance between perpendicular segments.

Category 2: Segment  $i$  and  $j$  are parallel to each other but are on the opposite sides of the spiral.

$$Z_{ij}^L = -j\omega M_{ij} \quad (7)$$

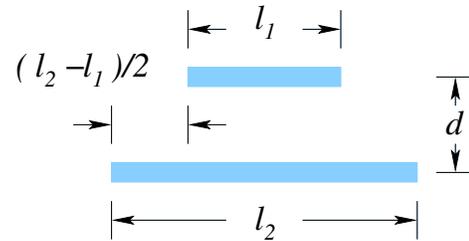
where  $M_{ij}$  is the mutual inductance between segment  $i$  and  $j$ . For the two wire segments shown in Fig. 4, the mutual inductance is given by

$$M_{ij} = M\left(\frac{l_1 + l_2}{2}, d\right) - M\left(\frac{l_2 - l_1}{2}, d\right) \quad (8)$$

where

$$M(l, d) = \frac{\mu l}{2\pi} \left[ \ln\left(\frac{\sqrt{l^2 + d^2} + l}{d}\right) - \frac{\sqrt{l^2 + d^2}}{l} + \frac{d}{l} \right] + \frac{\mu l}{2\pi} \left[ \frac{w^2 l}{12d^2 \sqrt{l^2 + d^2}} - \frac{t^2 l}{12d^2 (d + \sqrt{l^2 + d^2})} \right] \quad (9)$$

with  $w$  and  $t$  representing the width and thickness of the metal segments, respectively [8].



**Figure 4. Configuration of the two metal segments for mutual inductance computation**

Category 3: Segment  $i$  and  $j$  are parallel to each other and are on the same side of the spiral.

Generally, there is no simple closed form formula available

in this case for either the mutual inductance between segment  $i$  and  $j$  or the self inductance of a segment because of the presence of the skin effect and proximity effect. These two high frequency effects result in a nonuniform current distribution across the cross section of the metal segment which invalidates all the formulas based on the assumption that the current distribution is uniform. As pointed out in [9], the proximity effect may become significant at frequencies as low as a few hundred MHz for integrated spiral inductors, which means that it should be considered in all simulations of radio frequency devices.

To take care of the nonuniform current distribution in the metal segments due to skin and proximity effects, each metal segment is first divided into filaments along the longitudinal direction and it is assumed that the current distribution in each filament is uniform [5] [10]. Assume we have  $n$  parallel metal segments on each side of the spiral, the  $i^{th}$  segment is divided into  $N_i$  filaments, and let  $(i, j)$  represents the  $j^{th}$  filament of segment  $i$ , we have

$$\begin{aligned} V_{ij} &= \sum_{k=1}^n \sum_{m=1}^{N_k} (r_{ij} \delta_{ki} \delta_{jm} + j\omega M_{ij,km}) I_{km} \\ &= \sum_{k=1}^n \sum_{m=1}^{N_k} Z_{ij,km} I_{km} \end{aligned} \quad (10)$$

where  $V_{ij}$  is the voltage across filament  $(i, j)$ ,  $r_{ij}$  is the DC resistance of filament  $(i, j)$ , and  $M_{ij,km}$  is the mutual inductance between filament  $(i, j)$  and  $(k, m)$  which can be computed using (8) and (9). If  $(i, j) = (k, m)$  however,  $M_{ij,km}$  becomes the self inductance of filament  $(i, j)$  and should be computed using

$$L = \frac{\mu l}{2\pi} \left[ \ln\left(\frac{2l}{w+t}\right) + 0.5 + \left(\frac{\sqrt{w^2 + t^2 + 0.46tw}}{3l}\right) - \frac{w^2 + t^2}{24l^2} \right] \quad (11)$$

where  $l$ ,  $w$ , and  $t$  are the length, width, and thickness of the filament, respectively [8]. Equation (10) can be inverted to get

$$I_{ij} = \sum_{k=1}^n \sum_{m=1}^{N_k} Y_{ij,km} V_{km} \quad (12)$$

Since  $V_{km}$  are the same for the filaments in the same segment, the total current in the  $i^{th}$  segment can be computed using

$$\begin{aligned} I_i &= \sum_{j=1}^{N_i} I_{ij} = \sum_{j=1}^{N_i} \sum_{k=1}^n \sum_{m=1}^{N_k} V_k Y_{ij,km} \\ &= \sum_{k=1}^n \left( \sum_{j=1}^{N_i} \sum_{m=1}^{N_k} Y_{ij,km} \right) V_k = \sum_{k=1}^n Y_{ik} V_k \end{aligned} \quad (13)$$

Equation (13) can be inverted again to get

$$V_i = \sum_{k=1}^n Z_{ik} I_k \quad (14)$$

The  $Z_{ik}$ 's should be used to construct the portions of the  $Z^L$  matrix corresponding to the parallel segments on the same side of the spiral.

### 3.2 Sensitivity computation

To apply the SQP method, we must be able to compute the sensitivity of the inductance value  $L$  and quality factor  $Q$  with respect to each optimization variable. Let  $p$  represent an optimization variable, then

$$\frac{\partial L}{\partial p} = \frac{1}{2\pi f} \text{Im}\left(\frac{1}{Y_{12}^2} \frac{\partial Y_{12}}{\partial p}\right) \quad (15)$$

and

$$\frac{\partial Q}{\partial p} = -\frac{\text{Im}\left(\frac{\partial Y_{11}}{\partial p}\right) \text{Re}(Y_{11}) - \text{Im}(Y_{11}) \text{Re}\left(\frac{\partial Y_{11}}{\partial p}\right)}{[\text{Re}(Y_{11})]^2} \quad (16)$$

From (15) and (16), we can see that to obtain the sensitivity, we must compute the partial derivative of the  $Y$  parameters with respect to the optimization variables. One way to accomplish this is to use the finite difference approximation which requires one extra simulation for the sensitivity with respect to each optimization variable. Since simulation is an expensive process, we chose to use the adjoint method to compute the sensitivity which does not require any extra simulation at all.

Since the  $Y$  parameters can be obtained from  $\mathbf{I}_V$  in (2) by setting  $V_1$  and  $V_2$  to appropriate values, the question comes down to the computation of the sensitivity of  $\mathbf{I}_V$  with respect to the optimization variables. Note that the value of the optimization variables only affect the coefficient matrix in (2) but not the right hand side, we consider the following generalized system of linear equations.

$$A\mathbf{x} = \mathbf{b} \quad (17)$$

where  $A$  is the coefficient matrix with  $p$  as the parameter and  $\mathbf{b}$  is a vector independent of  $p$ , then we have

$$\frac{\partial x_i}{\partial p} = \sum_{ij} \frac{\partial x_i}{\partial A_{ij}} \frac{\partial A_{ij}}{\partial p} \quad (18)$$

$\frac{\partial A_{ij}}{\partial p}$  can be obtained directly from the physical model used to construct the coefficient matrix, and  $\frac{\partial x_i}{\partial A_{ij}}$  can be computed using the adjoint method [11], which gives

$$\frac{\partial x_i}{\partial A_{kl}} = -\xi_{ik} x_l \quad (19)$$

where  $\xi_{ik}$  is the  $k^{th}$  element of vector  $\xi_i$  and  $\xi_i$  is the solution to the equation

$$A^T \xi_i = \mathbf{e}_i \quad (20)$$

Here  $\mathbf{e}_i$  is the  $i^{th}$  column of the identity matrix. Note that (20) is extremely cheap to solve because after solving (17) using LU factorization, the LU factors of  $A^T$  can be obtained

	Case 1		Case 2		Case 3		Case 4	
<i>Lexp</i>	4.5nH		6nH		12nH		15nH	
<i>Freq</i>	2GHz		2GHz		2GHz		1GHz	
<i>D bound</i>	150μm/250μm		200μm/400μm		600μm/800μm		600μm/800μm	
<i>w bound</i>	2μm/10μm		2μm/20μm		2μm/20μm		2μm/20μm	
<i>s bound</i>	2μm/10μm		2μm/10μm		2μm/20μm		2μm/20μm	
<i>Num. turns</i>	3		3		3		3	
<i>Opt. Result</i>	Enum	SQP	Enum	SQP	Enum	SQP	Enum	SQP
<i>D</i>	250μm	250μm	400μm	400μm	600μm	600μm	780μm	778.42μm
<i>w</i>	8μm	8.18μm	16μm	17.55μm	10μm	10.92μm	20μm	20μm
<i>s</i>	2μm	2μm	4μm	2μm	4μm	3.53μm	2μm	2μm
<i>L</i>	4.32nH	4.28nH	5.77nH	5.70nH	11.73nH	11.40nH	14.28nH	14.25nH
<i>Q</i>	7.37	7.46	10.53	10.78	6.92	6.96	7.37	7.38
<i>Runtime</i>	170s	11s	2286s	41s	4579s	28s	2179s	22s

**Table 1. Comparison of the optimization results using SQP and enumeration**

automatically, i.e.,  $A^T = U^T L^T$  if  $A = LU$ , and the cost of solving (20) is simply one forward/backward substitution.

## 4 SQP algorithm

Sequential quadratic programming is a versatile method for solving the general nonlinear constrained optimization problems of the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \end{aligned} \quad (21)$$

where  $f : R^n \rightarrow R$ ,  $\mathbf{h} : R^n \rightarrow R^m$ , and  $\mathbf{g} : R^n \rightarrow R^p$ . Our formulation of the inductor optimization problem fits perfectly into the framework of (21). The Lagrangian of the problem is defined as

$$L(\mathbf{x}, \mathbf{u}, \mathbf{v}) = f(\mathbf{x}) + \mathbf{u}^t \mathbf{h}(\mathbf{x}) + \mathbf{v}^t \mathbf{g}(\mathbf{x}) \quad (22)$$

In each iteration of the algorithm, a quadratic subproblem

$$\begin{aligned} & \text{minimize} && \nabla f(\mathbf{x}^k)^t \mathbf{d}_x + \frac{1}{2} \mathbf{d}_x^t B_k \mathbf{d}_x \\ & \text{subject to} && \nabla \mathbf{h}(\mathbf{x}^k)^t \mathbf{d}_x + \mathbf{h}(\mathbf{x}^k) = \mathbf{0} \\ & && \nabla \mathbf{g}(\mathbf{x}^k)^t \mathbf{d}_x + \mathbf{g}(\mathbf{x}^k) \leq \mathbf{0} \end{aligned} \quad (23)$$

is formed where  $B_k$  is the approximation to the Hessian matrix  $H_L$  of the Lagrangian with respect to  $\mathbf{x}$ . The gradients  $\nabla f$ ,  $\nabla \mathbf{h}$ , and  $\nabla \mathbf{g}$  are computed using the sensitivity analysis discussed in section 3.2. The quadratic subproblem can be solved efficiently using any well known method such as the active set method. If  $\mathbf{d}_x^k$  is the solution to (23) in iteration  $\mathbf{k}$ , then the solution to the original problem (21) can be updated using

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha \mathbf{d}_x^k \quad (24)$$

where  $\alpha$  is the step length parameter. As pointed out in [12],

there are different ways that the Hessian matrix of the Lagrangian can be approximated, one of the most popular ones is due to Broyden, Fletcher, Goldfarb, and Shanno, where  $B_k$  is initially set to the identity matrix and updated using the formula

$$B_{k+1} = B_k + \frac{\mathbf{y}_k \mathbf{y}_k^t}{\mathbf{s}_k^t \mathbf{y}_k} - \frac{B_k \mathbf{s}_k \mathbf{s}_k^t B_k}{\mathbf{s}_k^t B_k \mathbf{s}_k} \quad (25)$$

where

$$\mathbf{s}_k = \mathbf{x}^{k+1} - \mathbf{x}^k \quad (26)$$

$$\mathbf{y}_k = \nabla_{\mathbf{x}} L(\mathbf{x}^{k+1}, \mathbf{u}^{k+1}, \mathbf{v}^{k+1}) - \nabla_{\mathbf{x}} L(\mathbf{x}^k, \mathbf{u}^k, \mathbf{v}^k) \quad (27)$$

To ensure the positive definiteness of  $B_k$ , the  $\mathbf{y}_k$  vector is reset to

$$\mathbf{y}_k \leftarrow \theta_k \mathbf{y}_k + (1 - \theta_k) B_k \mathbf{s}_k \quad (28)$$

if  $\mathbf{y}_k^t \mathbf{s}_k$  is not sufficiently positive. Here  $\theta_k \in [1, 0)$  is the number closest to 1 such that  $\mathbf{y}_k^t \mathbf{s}_k \geq \sigma \mathbf{s}_k^t B_k \mathbf{s}_k$  for some  $\sigma \in (0, 1)$  [13].

The iteration will continue until the Karush-Kuhn-Tucker condition is satisfied and the  $\mathbf{d}_x$  vector becomes zero. We have only provided the most basic background of the SQP algorithm here and will not elaborate on this topic any further. Interested readers are referred to [12] and [14].

## 5 Experiment results

We used the CFSQP package [15] as the SQP optimization engine to optimize the quality factor of square spiral inductors. The inductors are assumed to be fabricated using  $1\mu\text{m}$  thick metal with sheet resistance of  $20\text{m}\Omega/\square$  and rest  $5\mu\text{m}$  above the substrate. The lower and upper bound of the optimization variables are provided to the optimizer as the input and the tolerance of the allowed inductance deviation is set to 5%. Table 1 compares the optimization result of four sample inductors using SQP and enumeration.  $L_{exp}$  and  $Freq$  are the expected inductance and operating frequency of the de-

vice.  $D$  bound,  $w$  bound, and  $s$  bound specify the lower and upper bound of the outer length, trace width, and the space between metal traces, respectively. For the SQP method, the initial solution provided to the optimizer is such chosen that each variable takes the average value of its lower and upper bound. If this solution is infeasible, the SQP algorithm will first find a feasible solution automatically and then start optimization [15]. For the enumeration method,  $D$ ,  $w$ , and  $s$  are incremented by  $10\mu\text{m}$ ,  $2\mu\text{m}$ , and  $2\mu\text{m}$  respectively each time.

From the comparison result, we can see that the quality of the optimized design obtained from SQP is as good as that from enumeration while the runtime of the former is at least one order of magnitude smaller than that of the latter. We point out that the grid we used for the enumeration method is quite coarse, and if finer grid is used, the advantage of SQP over enumeration will become even more significant. In addition, we would like to emphasize that the goal of this work was to demonstrate the speedup we can obtain by using the SQP instead of the enumeration algorithm. The code for extracting the inductance and quality factor is not optimized for speed and accuracy. However, even if a different program is used to implement the extraction engine, we can still expect SQP to have similar speedup ratio compared with enumeration.

A subtle point concerning the SQP algorithm versus enumeration is that the optimization problem we are solving has not been proven to be a convex program. As a result, no mathematical programming algorithm can guarantee the finding of the global optimum, i.e., there is a possibility that the algorithm will stop at a local optimum. However, extensive experiments have shown that most of the time, the global optimum can be found in a single run of the SQP algorithm, and in case the algorithm does stop at a local optimum in one run, we can still obtain the global optimum by running the algorithm at most a few times starting from different initial points. This will preserve the validity of our claim that the SQP algorithm is superior to enumeration in terms of runtime considering the order of magnitude speedup each single run of SQP can achieve.

## 6 Conclusions

In this work, we used sequential quadratic programming to optimize the quality factor of integrated spiral inductors. Experiment results demonstrated that SQP can achieve at least an order of magnitude speedup compared with enumeration while maintaining the same quality of the optimized design. Besides its high local convergence rate, another advantage of the SQP method is that it makes no assumption about the formality of either the objective or the constraint functions, which makes it quite compatible with physical models derived from first principles.

## References

- [1] C. P. Yue and S. S. Wong, "On-chip Spiral Inductors with Patterned Ground Shields for Si-based RF IC's," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 5, pp. 743-752, May 1998.
- [2] J. Y. C. Chang, A. A. Abidi, and M. Gaitan, "Large Suspended Inductors on Silicon and Their Use in a  $2\text{-}\mu\text{m}$  CMOS RF amplifier," *IEEE Electronic Device Letter*, vol. 14, pp. 246-248, May 1993.
- [3] A. M. Niknejad, "Analysis, Simulation, and Applications of Passive Devices on Conductive Substrate," Ph. D. Thesis, Spring 2000.
- [4] M. D. M. Hershenson, S. S. Mohan, S. P. Boyd, and T. H. Lee, "Optimization of Inductor Circuits via Geometric Programming," *IEEE/ACM Proceedings of the 36<sup>th</sup> Design Automation Conference*, pp. 994-998, 1999.
- [5] A. M. Niknejad and R. G. Meyer, "Analysis, Design, and Optimization of Spiral Inductors and Transformers for Si. RF IC's," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 10, pp. 1470-1481, Oct. 1998.
- [6] S. C. Wong, G. Y. Lee, and D. J. Ma, "Modeling of Interconnect Capacitance, Delay, and Crosstalk in VLSI," *IEEE Transactions on Semiconductor Manufacturing*, vol. 13, no. 1, pp. 108-111, Feb. 2000.
- [7] K. Nabors and J. White, "FastCap: A Multipole Accelerated 3-D Capacitance Extraction Program," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 10, no. 11, pp. 1447-1459, Nov. 1991.
- [8] S. S. Mohan, "The Design, Modeling, and Optimization of On-chip Inductor and Transformer Circuits," Ph. D. Thesis, Dec. 1999.
- [9] W. B. Kuhn and N. M. Ibrahim, "Analysis of Current Crowding Effects in Multiturn Spiral Inductors," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 1, pp. 31-37, Jan., 2001.
- [10] W. T. Weeks, L. L. Wu, M. F. McAllister, and A. Singh, "Resistive and Inductive Skin Effect in Rectangular Conductors," *IBM J. Res. Develop.*, vol. 23, pp. 652-660, Nov. 1979.
- [11] L. T. Pillage, R. A. Rohrer, and C. Visweswariah, "Electronic Circuit and System Simulation Methods," pp. 67-74, McGraw-Hill, New York, 1995.
- [12] P. T. Boggs and J. W. Tolle, "Sequential Quadratic Programming," *Acta Numerica*, 1996.
- [13] "Sequential Quadratic Programming," [http://www-fp.mcs.anl.gov/otc/Guide/OptWeb/continuous/constrained/nonlinearcon/section2\\_1\\_1.html](http://www-fp.mcs.anl.gov/otc/Guide/OptWeb/continuous/constrained/nonlinearcon/section2_1_1.html).
- [14] J. Nocedal and S. J. Wright, "Numerical Optimization," pp. 528-575, Springer, New York, 1999.
- [15] C. Lawrence, J. L. Zhou, and A. L. Tits, "User's Guide for CF-SQP Version 2.5," <http://64.238.116.66/aemdesign/download-cfsqp/cfsqp-manual.pdf>.