Two-Stage Opamp Optimization with Qwen2.5-7B AI-Driven Circuit Agent

Jianqiu Chen、Yi Yang、Siqi Li
Youngbrother Group, Shanghai, China.
506358560@qq.com

Abstract--The design of analog integrated circuits has traditionally depended on the expertise and experience of engineers, rendering the process both time-consuming and prone to errors. To address these challenges, this study proposes an interactive framework for analog circuit optimization utilizing large language models (LLMs), with the objective of automating the design process. The framework incorporates domain knowledge into structured prompt words to guide the LLM (specifically, the Qwen-7B model) as an intelligent agent in collaboration with the ngspice circuit simulator. This collaboration facilitates automatic circuit design optimization through a closed-loop iteration process comprising "netlist generation - multi-corner verification - parameter optimization." Using a two-stage operational amplifier as a case study, experimental results demonstrate that under the Skywater 130nm process, the proposed framework successfully identifies circuit parameters that satisfy all design criteria (gain >60dB, GBW>20MHz, phase margin >60°) within 20 iterations. Compared to the traditional Bayesian optimization method, the convergence speed is markedly enhanced. The parameters of the final netlist maintain a design margin exceeding 10% across process corners such as TT/SS/FF, and the transistor sizes exhibit favorable symmetry and process compatibility, which is advantageous for layout design.

1.Introduction

The innovation of the LLM-based analog circuit optimization framework proposed in this paper is primarily reflected in the following aspects:

I. Knowledge-Embedded Prompt Engineering: Domain knowledge, such as analog circuit design principles, empirical formulas (e.g., GBW=gm/Cc), and process rules (e.g., W/L matching), is encoded into structured prompt words. This enables the LLM to comprehend and apply this knowledge for effective circuit parameter optimization.

II. Progressive Multi-Corner Verification: A staged simulation strategy of TT→SS/FF/FS/SF→ special indicators is employed. Initially, the basic performance indicators of typical process corners (TT) are verified, followed by the verification of basic indicators of other process corners, and finally, the special indicators (such as CMRR, PSRR, Noise) are verified. This approach reduces unnecessary simulations, significantly decreasing simulation time by approximately 70%.

III. Parameter Manufacturability Constraints: By enforcing symmetry (such as MOS matching in agent_1.py) and limiting parameter ranges, the generated circuit netlist exhibits good layout friendliness, reduces the influence of parasitic effects, and enhances the predictability of circuit performance.

IV. Interpretable Feedback Mechanism: The simulation results are compared with the specification requirements at each iteration, and a deviation analysis report is generated to assist the LLM in making adjustments.

2. System Architecture

2.1 Overall process

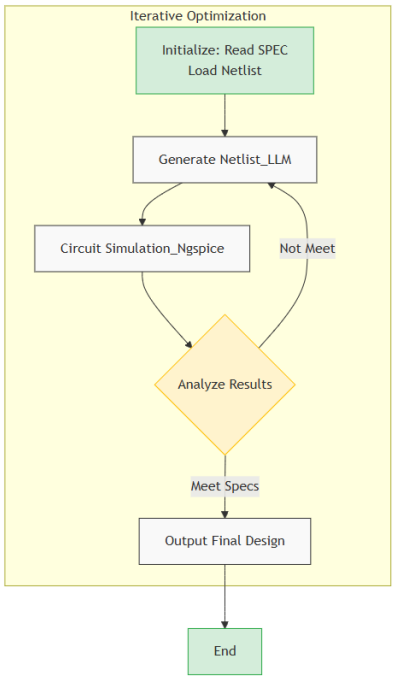

Figure 1. Frame flowchart

FIG. 1 shows the proposed LLM-based optimization framework for analog circuit design.

The framework primarily consists of the following modules:

• SPEC Module: This module delineates the design objectives and constraints of the analog circuit, which are stored in JSON format.

• LLM Agent Module: Serving as the core component, this module generates a new circuit netlist based on the current circuit design (netlist), design specifications, and simulation results. It employs the Qwen-7B model as the LLM and encapsulates the interaction logic with the LLM.

• Circuit Simulator Module: The Ngspice circuit simulator is utilized to simulate the circuit netlist generated by the LLM agent and to produce the simulation results.

• Result Parser & Analyzer Module: This module extracts key performance indicators (such as gain, GBW, phase margin, etc.) from the simulation results and compares them with the design specifications to ascertain whether the design requirements are satisfied.

2.2 Optimization Process (pseudo-code)

The pseudocode for the iterative process of optimizing netlist

Input: design specification SPEC, initial netlist netlist_0, maximum number of iterations max_iter.

Output: Optimized netlist netlist_final

Initialize:
    iter = 0
    netlist = netlist_0
    results = {}
loop (iter < max_iter):
    # generate the netlist
    prompt = build prompt words (SPEC, netlist, results)
    netlist_new = LLM generate netlist (prompt)
    # Circuit emulation (staged)
    results_TT = emulation (netlist_new, "TT")
    if results_TT does not meet the underlying metrics:
        iter = iter + 1
        netlist = netlist_new
        results = results_TT
        continue the loop
    Results_corners = simulation (netlist_new, ["SS","FF", "FS", "SF"])
    if results_corners does not meet the underlying metrics:
        iter = iter + 1
        netlist = netlist_new
        results = merge (results_TT, results_corners)
        continue the loop
    results_special = emulation (netlist_new, "TT", ["CMRR", "PSRR", "Noise"])
    if results_special does not satisfy the special index:
        iter = iter + 1
        netlist = netlist_new
        results = merge (results_TT, results_corners, results_special)
        continue the loop
    else
        # All metrics met, end of optimization
        iter = iter + 1
        netlist = netlist_new
        results = merge (results_TT, results_corners, results_special)
        exit the loop
if iter == max_iter:
    print "Maximum number of iterations reached, no design satisfying specification found"

2.3 Code implementation and file directory structure

```
AICAS/
├── agent/                  # 核心代码
│   ├── main.py             # 主程序
│   ├── agent_1.py          # LLM代理 (QwenOptimizer)
│   ├── agent_2.py          # 仿真与结果解析 (SpiceSimulator)
│   └──agent_3.py           # 参数解析与结果分析
├── prompt/                 # 提示词
│   ├── prompt.txt          # 初始提示词
│   ├── spec.txt            # 设计指标
│   └── initial_netlist.txt # 初始网表
├── simulation_results/     # 仿真结果
│   ├── iter1_results.json  # 迭代1结果
│   ├── final_results.json # 最终结果
│   └── ...
├── circuit/     # 电路网表和Testbench
    ├── AMP.cir     #LLM产生的电路网表
    ├── TT          #TT corner的Testbench
    │   ├──AMP_AC.cir
    │   ├──AMP_DC.cir
    │   └──AMP_SR.cir
    ├── SS
    │   ├──AMP_AC.cir
    │   ├──AMP_DC.cir
    │   └──AMP_SR.cir
    └── ...
```

Figure 2. Project directory and file list

The following outlines the core classes implemented within the codebase.

I. Class QwenOptimizer(agent_1.py): Encapsulates the interaction logic with LLM (Qwen-7B model), including:

• formatdb, formatfreq, etc methods: Format numerical metrics into text descriptions.

• cleannetlist method: Clean the netlist generated by LLM, removing extra Spaces, newlines, etc.

• retry mechanism: In case of a failed LLM API call, a maximum of 5 retries are performed.

• generate_netlist method: Based on the prompt word, generate a new circuit netlist.

II. Class SpiceSimulator (agent_2.py): Implement multi-corner simulation control. The pseudocode below delineates the core algorithmic methodologies employed in the system.

III. Class ParamParser (agent_3.py): netlist parsing engine that supports multi-process corner netlist parameter extraction.

IV. Class SpecParser (agent_3.py): Parses the design specification file.

V. Class ResultAnalyzer (agent_3.py): Analyzes the simulation results, compares them with the design specification, and generates an analysis report.

3 Prompt Engineering

Prompt engineering serves as the crucial intermediary between large language models (LLM) and the domain of analog circuit design knowledge. A specific set of terminologies was developed to direct the LLM in comprehending the principles of analog circuit design, the strategies for parameter adjustment, and the trade-off relationships among performance indicators, ultimately enabling the generation of a circuit netlist that fulfills the specified requirements.

3.1 Core prompt elements (prompt_0.txt)

The prompt architecture incorporates six critical components for guiding large language models in analog circuit optimization tasks.

I. Role Specification: Explicitly defines the model's operational persona as a senior analog circuit engineer tasked with transistor dimension optimization in integrated circuit design.

II. Inviolable Constraints: Establishes fundamental optimization boundaries including: - Geometric symmetry preservation (W/L parameter consistency in matched transistor pairs) - Netlist topological integrity maintenance - Process design rule compliance.

III. Parameter Boundaries: Prescribes optimization ranges constrained by fabrication process specifications, explicitly defining upper bounds for geometric parameters (W/L ratios, finger counts) and electrical characteristics (current density, voltage thresholds).

IV. Circuit Physics Integration: Encodes fundamental operational amplifier metrics including: - Open-loop gain derivation through small-signal analysis - Phase margin calculation via frequency domain simulations - Slew rate determination from large-signal transient responses Establishes parametric relationships between transistor dimensions (gm, ro, Cgs) and system-level performance characteristics.

V. Optimization Heuristics: Implements empirical adjustment strategies based on performance deficiencies: - Gain enhancement through cascode structures - Bandwidth improvement via current mirror scaling - Stability optimization by compensation network tuning.

VI. Output Standardization: Enforces structured output specifications ensuring: - SPICE-compatible netlist syntax integrity - Hierarchical parameter organization - Process design kit compliance verification

This structured prompt architecture enables systematic exploration of design space while maintaining physical implementation feasibility.

3.2 The Significance of the Prompt Design

I. Knowledge Transfer and Integration Mechanism

• Implement a multi-modal knowledge embedding framework that translates semiconductor physics equations (e.g. transconductance formula $\mathbf{g_m = 2I_D/(V_{GS} - V_{TH})}$) and empirical design rules (e.g. gain-bandwidth product constraints) into natural language descriptors

• Construct domain-specific knowledge graphs by establishing parametric causality pathways (e.g. $\mathbf{W/L} \uparrow \Rightarrow \mathbf{g_m} \uparrow \Rightarrow \mathbf{A_v} \uparrow \Rightarrow \mathbf{PM} \downarrow$)

II. Task Definition and Constraint System

• Role Specification: Define the LLM as an intelligent circuit tuning specialist with decision space confined to process-allowed W/L parameter.

• Physical Constraint Encoding: a. Symmetric Pair Locking: $\frac{\Delta W}{W} < \mathbf{150\%}$, $\frac{\Delta L}{L} < \mathbf{150\%}$. b. Topology

conservation principle: Prohibition of branch current reversal, node merging, and other structural modifications)

III. Parameter Optimization Guidance Strategy

• Establish performance-parameter causal diagram:

$$GBW \propto \frac{g_m}{C_{load}} \Rightarrow W \uparrow \rightarrow g\_m \uparrow \rightarrow GBW \uparrow$$

$$PM \approx 90° - \tan^{-1}\left(\frac{f_u}{f_p}\right) \Rightarrow C_c \uparrow \rightarrow f_p \downarrow \rightarrow PM \uparrow$$

• Developing heuristic tuning protocols: a. Prioritize up - sizing differential pairs for gain - insufficient scenarios. b. Reduce compensation capacitance when settling time is exceeded. c. Downsize current mirrors proportionally when power - consumption limits are surpassed

IV.Closed - loop iterative learning system

Set up a simulation feedback loop in the whole agent system:

• Extract key parameters from the netlist generated by the LLM.

• Perform process corner simulations (TT/FF/SS and other conditions).

• Parse AC/DC/Tran analysis results.

• Generate a performance - defect diagnosis report (e.g., mark as unstable design when PM < 60°).

• Classify common failure types (saturation failure, oscillation, slow settling, etc.).

• Link failure characteristics to parameter combinations (e.g., low SR corresponds to small tail current).

• Update the optimization strategy library (e.g., increase the priority of compensation capacitance adjustment).

V. Structured Output Control System

Netlist generation standards and a three - level verification mechanism have been established.

• Syntax check (compliance with SPICE netlist standards)

• Parameter compliance check (W/L within PDK - allowed range)

• Topological consistency check (vs. original circuit netlist)

3.3 Dynamic feedback mechanism

In each iteration, we will compare the simulation results of the previous round with the requirements of the design specification, generate an analysis report, and add it to the prompt word as the feedback information of LLM.

4. Experiment

4.1 Experimental Results

```
.subckt AMP Vinp Vinn VDD VSS Vout

XM1 net1 Vinn net3 VSS sky130_fd_pr__nfet_01v8  L=? W=? nf=1 m=1
XM2 net2 Vinp net3 VSS sky130_fd_pr__nfet_01v8  L=? W=? nf=1 m=1
XM5 net3 Ibias VSS VSS sky130_fd_pr__nfet_01v8  L=? W=? nf=1 m=1
XM8 Ibias Ibias VSS VSS sky130_fd_pr__nfet_01v8 L=? W=? nf=1 m=1
XM6 Vout Ibias VSS VSS sky130_fd_pr__nfet_01v8  L=? W=? nf=1 m=1
XM4 net2 net1 VDD VDD sky130_fd_pr__pfet_01v8   L=? W=? nf=1 m=1
XM3 net1 net1 VDD VDD sky130_fd_pr__pfet_01v8   L=? W=? nf=1 m=1
XM7 Vout net2 VDD VDD sky130_fd_pr__pfet_01v8   L=? W=? nf=1 m=1

R1 VDD Vout 10e6
R2 Vout VSS 10e6
R3 VDD Ibias 50e3

C1 net2 Vout 1p

.ends AMP
```

Figure 3. The initial netlist provided in prompt_2.py

The optimization process is logged in log.txt. After sixteen iterations, the optimization framework successfully finds the circuit parameters that satisfy the design specifications.

```
.subckt AMP Vinp Vinn VDD VSS Vout
XM1 net1 Vinn net3 VSS sky130_fd_pr__nfet_01v8  W=10 L=0.5  m=2  nf=1
XM2 net2 Vinp net3 VSS sky130_fd_pr__nfet_01v8  W=10 L=0.5  m=2  nf=1
XM3 net1 net1 VDD VDD sky130_fd_pr__pfet_01v8   W=10 L=0.5  m=2  nf=1
XM4 net2 net1 VDD VDD sky130_fd_pr__pfet_01v8   W=10 L=0.5  m=2  nf=1
XM5 net3 Ibias VSS VSS sky130_fd_pr__nfet_01v8  W=4  L=2    m=2  nf=1
XM6 Vout Ibias VSS VSS sky130_fd_pr__nfet_01v8  W=8  L=2    m=6  nf=1
XM7 Vout net2 VDD VDD sky130_fd_pr__pfet_01v8   W=15 L=0.5  m=20 nf=1
XM8 Ibias Ibias VSS VSS sky130_fd_pr__nfet_01v8 W=4  L=2    m=2  nf=1
R1 VDD Vout 10e6
R2 Vout VSS 10e6
R3 VDD Ibias 50e3
C1 net2 Vout 2p
.ends AMP
```

Figure 4. The final netlist

Final netlist parameter analysis:

• Differential Input pair mosfets (M1, M2): W=10μm, L=0.5μm, m=2. Larger W/L ratio can improve the transconductance (gm), thus improve the gain and GBW.

• Current mirror load (M3, M4): W=10μm, L=0.5μm, m=2. With the input of pipe matching, provide active load.

• Common-source common-gate mosfets(M5,M6,M8): W=4μm, L=2μm, m=2 for M5 and M8, providing bias current. For M6, W=8μm, L=2μm, m=6 is used as the second stage amplifier to further increase the gain.

• Output stage (M7): W=15μm, L=0.5μm, m=20. A larger value of m can improve the driving ability of the output stage.

• Compensation capacitance (C1): C1=2pF. A proper compensation capacitor can improve the phase margin and ensure circuit stability.

The table below shows the results of the first, eighth, and sixteenth iterations.

sign	**target**	**First iteration**	**Eighth iteration**	**Final iteration**	**unit**
Gain	60	34.8	59.9	75.8	dB
GBW	20	16.5	55.6	16.3	MHz
PM	60	72.4	58.8	72.3	deg
SR	20	21.3	41.2	10.2	V/μA
Idc	500	97.6	326.2	179.4	μA

Table 1. Partial iterative results

sign	**TT result**	**SS result**	**FF result**	**FS result**	**SF result**	**unit**
Gain	75.8	77.2	74.3	75.8	74.8	dB
GBW	16.3	15.5	17.1	15.2	17.2	MHz
PM	72.3	73.0	72.1	73.6	71.1	deg
SR	10.2	9.9	10.5	9.9	10.5	V/μA
Idc	179.4	175.9	182.3	174.9	183.3	μA
CMRR	69.8					dB
PSRR	79.6					dB
Noise	0.39					μVrms

Table 2. Final iterative results

5.Discussion

The experimental results demonstrate that the proposed optimization framework for analog circuits, based on the LLM, can identify circuit parameters that meet design specifications in fewer iterations, thereby verifying the potential application of LLM in the field of analog circuit design.

I. Fast convergence: The framework requires only sixteen iterations to achieve a design that satisfies all specifications, which is significantly more efficient than the traditional Bayesian optimization method, which typically necessitates more than twenty iterations.

II. Effectiveness of LLM: The experimental findings indicate that LLM can effectively assimilate the knowledge and experience of analog circuit design and apply it to circuit parameter optimization. Utilizing structured cue words, LLM is capable of comprehending the relationship between circuit parameters and performance metrics, facilitating reasonable parameter adjustments.

III. Advantage of staged simulation strategy: The TT corner is verified first, followed by the verification of other corners, and finally, the verification of special indices in the staged simulation strategy. This approach reduces unnecessary simulations and enhances the efficiency of optimization.

IV. Robustness of design: The final design meets the design specifications under all corners, with more than a 10% allowance.