Microwave Journal

INVITED PAPER

From Ad Hoc Simulations to Orchestrated RF Design Workflows

Jason Boh and Matt Ozalas
Keysight Technologies

Don Dingee
STRATISET

The complexity of RF circuits and systems, coupled with stringent performance targets, has made simulation essential. Historically, simulations were configured and run manually, with engineers reviewing results and deciding next steps. More recently, Python-based automation has emerged as a compelling alternative, improving efficiency, scalability and repeatability. However, many RF engineers do not consider themselves programmers, making it difficult to translate their deep design expertise into workflows built entirely in code.

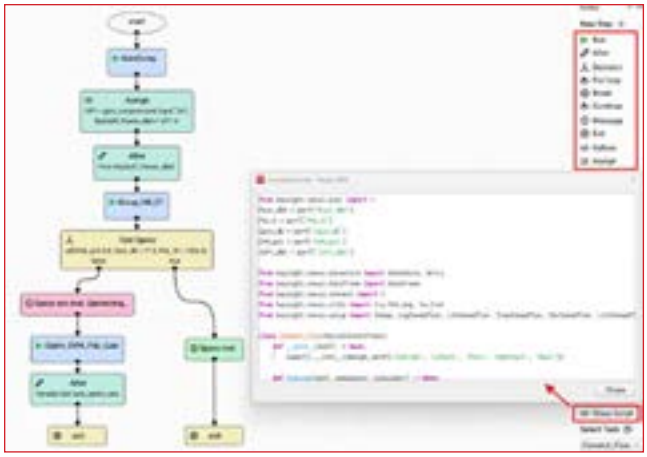
A new paradigm is emerging for orchestrating RF simulations. This methodology trades pure code-based automation for an executable visual flowchart. Yet it doesn't eliminate Python from the picture. Instead, the approach spans no-code, low-code and fully coded orchestration, adapting to both requirements and individual preferences. Combining Python with a visual flowchart offers the best of both worlds: humans can quickly understand the logic, while decision-making can be handed off to agentic AI.

We will demonstrate the benefits of a flowchart-based approach using an RF power amplifier (PA) design example and the technique of intrinsic node waveform engineering. This approach integrates common simulation tasks, including DCIV, S-parameter extraction, load- and source-pull analysis, harmonic balance (HB), parameter sweeping and optimization, into a single executable flow. Finally, we will discuss how AI can fit into this workflow through surrogate modeling, natural language prompting and agentic decision-making.

RF WORKFLOW ORCHESTRATION – A NEW FLOWCHART PARADIGM

Like many software applications, RF simulators began with command-line interfaces and later evolved into graphical user interfaces (GUIs). Dialog boxes are typically intuitive, with features such as buttons, checkboxes, drop-down menus and short text fields. The learning curve is short, but GUIs have downsides. For example, achieving consistent results across ad hoc simulation runs requires that all fields be entered in exactly the same way every time. Reusable simulation templates can help, but since they are usually editable, traceability remains elusive. Another major limitation of GUIs is scalability: as circuit

Reprinted with permission of MICROWAVE JOURNAL® from the July 2026 issue.
©2026 Horizon House Publications, Inc.



▲ Fig. 1 **A Nexus Connect flowchart and its associated Python code.**

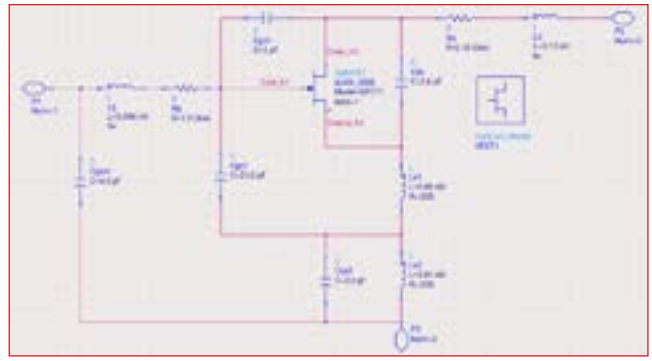
complexity and the number of design variables increase, manual editing becomes impractical.

Scripting improves the situation by fostering repeatability and enabling scalability. Many RF software tools began adding programmability through various (sometimes proprietary) scripting languages. Then Python burst onto the scene, growing into the world's most popular programming language. Some EDA vendors began efforts to retool around Python, benefiting from its vast ecosystem of open-source libraries and improved code portability and reuse.

Keysight leaned into the trend by adding Python support to its existing software platforms, including Advanced Design System (ADS). Simultaneously, research and development was underway on a next-generation, Python-native platform called RF Circuit Simulation Professional (Nexus), which has now been released and is available to customers. Nexus is designed with automation and AI-readiness in mind.[1]

Building on Nexus, Keysight is taking a new step in reimagining RF design workflows. Nexus Connect represents a new paradigm, enabling individuals and organizations to capture their design engineering processes as flowcharts, similar to how measurement processes are captured with visual tools. As shown in ***Figure 1***, the flowcharts comprise various action blocks, or "steps," underpinned by modifiable Python code. The flowchart manages tasks and decisions, allowing pre-planned, orchestrated workflows to run automatically and be reproduced precisely.

Visual flowchart diagrams help engineering teams capture and share information. From the moment a design project begins, engineering thinking becomes transparent and is saved electronically rather than lost when a physical whiteboard is erased. Instead of leafing through tedious documentation or tracking down colleagues, team members at any skill level can immediately see the types of simulations and parameters chosen, the goals used for optimization and the criteria for deciding how to proceed based on the results. Diagrams can also serve as the focal point for design reviews. New designs can start with reusable diagrams. The time savings from improved orchestration and reduced workflow friction add up quickly.



▲ Fig. 2 **An intrinsic-node GaN transistor model with a current generator and parasitics, represented in ADS.**

PA DESIGN WITH AN INTRINSIC NODE MODEL AND WAVEFORM ENGINEERING

The best way to illustrate this is with a real-world example. Here, we focus on RF PA design, specifically a Class J amplifier, which is similar to Class B operation but uses harmonic tuning to improve efficiency. The complexity of PA design makes it an ideal vehicle for discussing workflow orchestration. This discussion is kept at a high level so that all RF designers can follow along, not just those specializing in PAs. In this example, ADS is used for schematic capture and data display, and Nexus to run simulations.

Much has been written about extrinsic design methodologies for PAs, in part because they align with how designers measure and characterize circuits. This approach places the reference plane for simulations or measurements at a physically accessible boundary — typically the transistor terminals, either at the package leads or at the die edge. As a result, an extrinsic model or measured dataset inherently includes the parasitic resistance, capacitance and inductance present within the device.

In contrast, intrinsic design methodologies analyze voltages and currents at internal device nodes rather than at terminals. By working closer to the intrinsic current generator, designers can directly observe waveforms that are typically masked by device parasitics. This perspective provides deeper insight into transistor operation and enables the evaluation of key PA performance metrics. ***Figure 2*** shows a schematic of a conceptual, non-process-specific GaN transistor with its intrinsic device nodes explicitly exposed.

Access to intrinsic nodes allows PA designers to build circuits around first principles. The challenge with implementing an intrinsic-node method is that many manual steps are required to iteratively optimize a design.[2]

- Ideal nonlinear voltage and current waveforms are superimposed on a device's DCIV curves to determine the power and efficiency a device could achieve in that operating state.
- Strategically engineered waveforms reduce overlap between the time-domain voltage and current waveforms, minimizing resistive power dissipation.
- The Class J mode tunes the second-harmonic load impedance to boost the voltage waveform, while higher harmonics shape the current waveform, theoretically allowing an efficiency of 78.5 percent with a more flexible set of harmonic impedances than in

Class B.

- The time-domain waveforms are then transformed into the frequency domain using an FFT, yielding computed impedances at the fundamental and harmonic frequencies.
- By applying those impedances back to the idealized current generator in the intrinsic-node transistor model, energy organizes into the efficient waveforms we started with, maximizing achievable device power efficiency.

The PA design bench schematic in ADS, as seen in ***Figure 3***, mimics a load-pull measurement system.[3] It includes harmonic tuners, bias networks and receivers for the A (incident) and B (reflected) waves, enabling a complete analysis of PA metrics. The device under test (DUT) is the GaN FET model shown in Figure 2.

A single Nexus Connect task captures the sequence of analyses and tasks required to complete a significant portion of a Class J amplifier design. ***Figure 4*** shows the first few steps of the Nexus Connect executable flowchart.

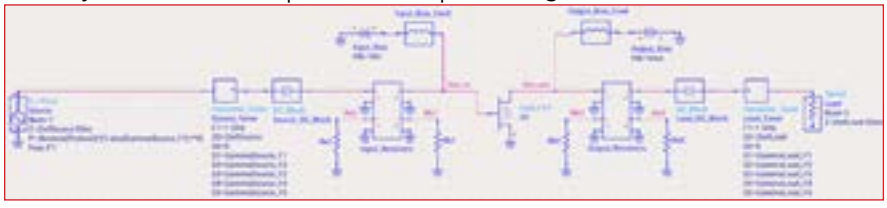


Fig. 3 **PA design bench schematic in ADS using the GaN FET model.**

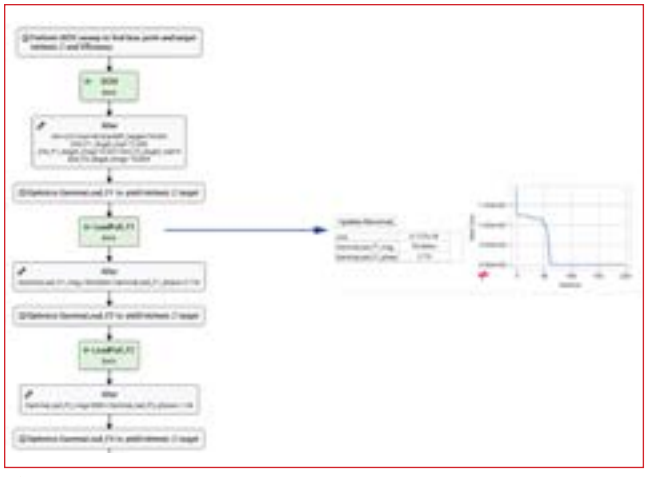


Fig. 4 **A Nexus Connect task execution flowchart with results of a load-pull optimization displayed.**

- First, the task performs a DCIV sweep, followed by load-line calculations for bias and target intrinsic impedances (Zs) at each harmonic.
- Then, it sequentially optimizes extrinsic loads for each harmonic to achieve the target intrinsic Zs.
- Next, it performs a source-pull analysis and an input power adjustment to match power and efficiency.
- Then, an S-parameter analysis is performed, including stability calculations with Winslow virtual probes.
- Finally, a power-swept HB analysis confirms the overall PA performance.

As the flowchart execution graph progresses, run blocks can change color (a programmable setting) and double-clicking a block can reveal simulation or optimization results. Figure 4 also shows a portion of the optimization results from the first load-pull analysis when the graph is clicked.

Designers can set top-level parameters such as the target output power, alpha (which reshapes the voltage waveform and changes the target impedances without affecting power or efficiency) and the percentage of the device's maximum output current for the quiescent (q) point. Re-running the flowchart with new top-level parameters produces updated results quickly (in less than a minute on a laptop).

Deriving target intrinsic gammas involves constructing a load line from the DCIV sweep data by connecting the q point to one of the knee points. First, simple Python algorithms and heuristics determine the q point from the device's breakdown voltage and the desired output current at cutoff (a small percentage of the maximum output current). Then, knee points are calculated for all simulated Vin curves, as shown in ***Figure 5***.

Next, load-line metrics are evaluated for all knee/quiescent point pairs, and additional Python algorithms are used to select a load line from the candidates. The algorithms balance output power and efficiency while walking down the knees. (The actual details of our algorithms are not the focus here — the key point is that all

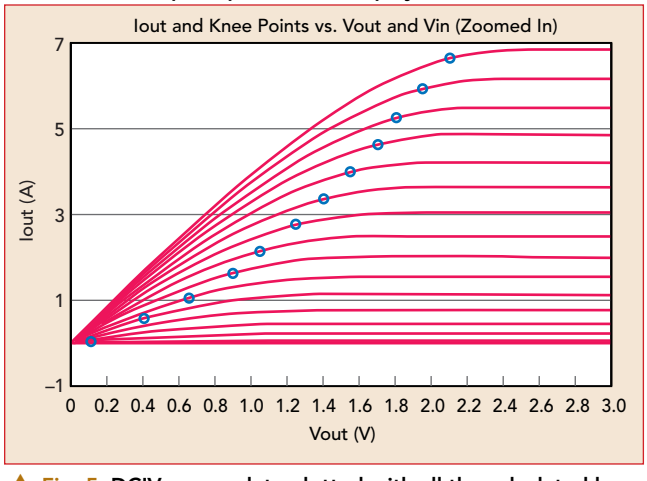


Fig. 5 **DCIV sweep data plotted with all the calculated knee points.**

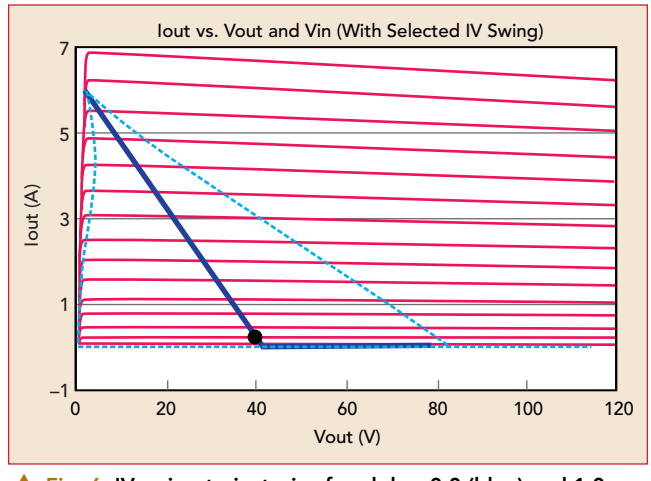
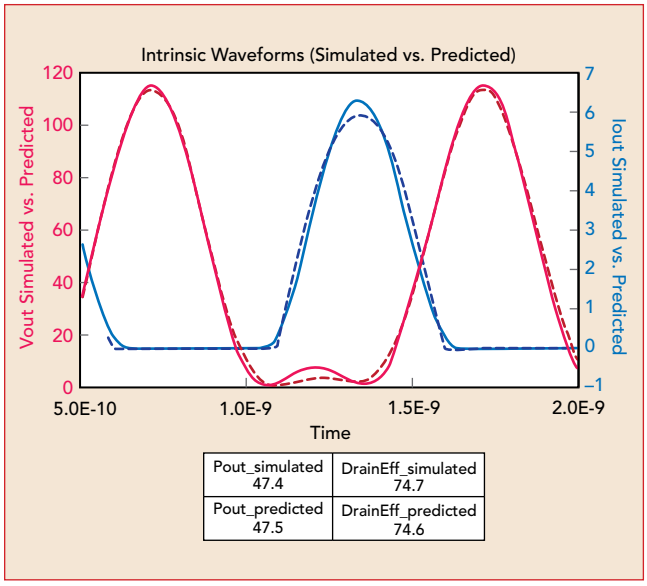


Fig. 6 **IV swing trajectories for alphas 0.0 (blue) and 1.0 (teal) and q point (black), calculated from DCIV sweep data.**



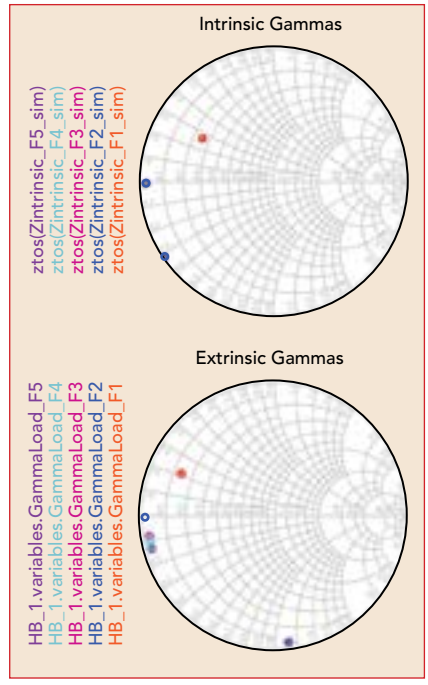
▲ Fig. 7 **Simulated (solid) versus predicted (dashed) intrinsic waveforms, output power and efficiency.**

decisions and rules are formally captured in the execution flowchart.) The blue line in ***Figure 6*** is the rectified load line (or IV swing at alpha equals 0.0). With the FET biased for Class J operation at the q point, the instantaneous current and voltage swing along the line when an RF signal is applied. The FET conducts zero current for half of the RF cycle, yielding a 50 percent duty cycle. With an alpha of 0.0, the voltage waveform is a pure sinusoid. Using alpha values of 1.0 or -1.0 gives a "harmonic boost" to the voltage waveform, moves the target intrinsic gammas off the real axis of the Smith chart and widens the IV swing trajectory away from the rectified load line shown in teal.

Once the Nexus Connect task is complete, we achieve a good match between the simulated and predicted intrinsic voltage and current waveforms, as well as the power and efficiency, as shown in ***Figure 7***.

Extrinsic gammas, which are ultimately needed for matching network design, differ from intrinsic gammas because of the parasitics in the device model. The Nexus Connect task sequentially optimizes the extrinsic gammas up to the fifth harmonic, see ***Figure 8***, to minimize errors in achieving the target intrinsic gammas.

The entire sequence shown, comprising simulations, algorithms and decisions, is captured and executed in a single Nexus Connect flowchart. While an engineer could have initiated the same steps manually via the Nexus graphical user interface, it would be hard to reproduce the exact repeatable steps from separate, disconnected analyses and tasks.

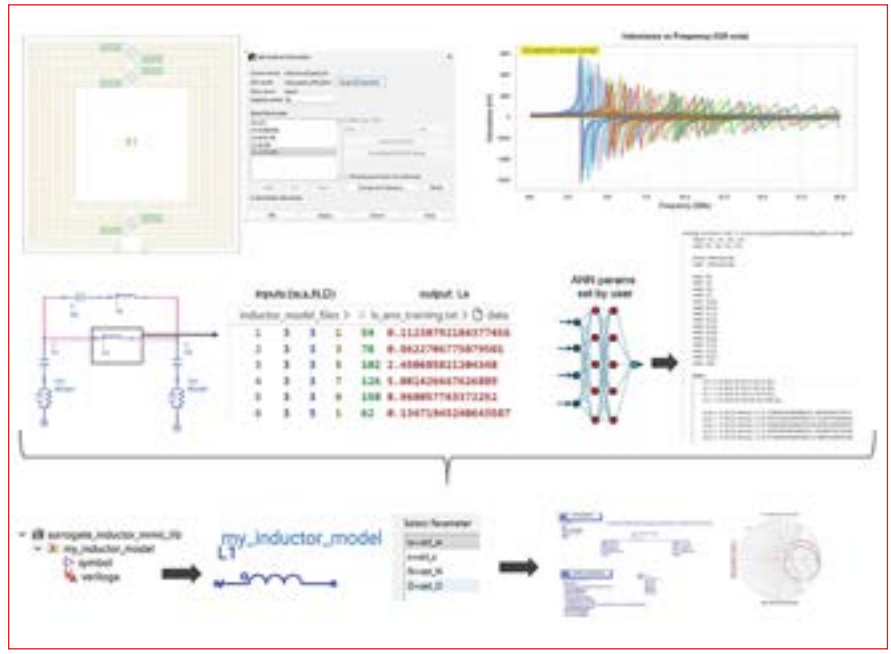


▲ Fig. 8 **Intrinsic and extrinsic gammas at each harmonic.**

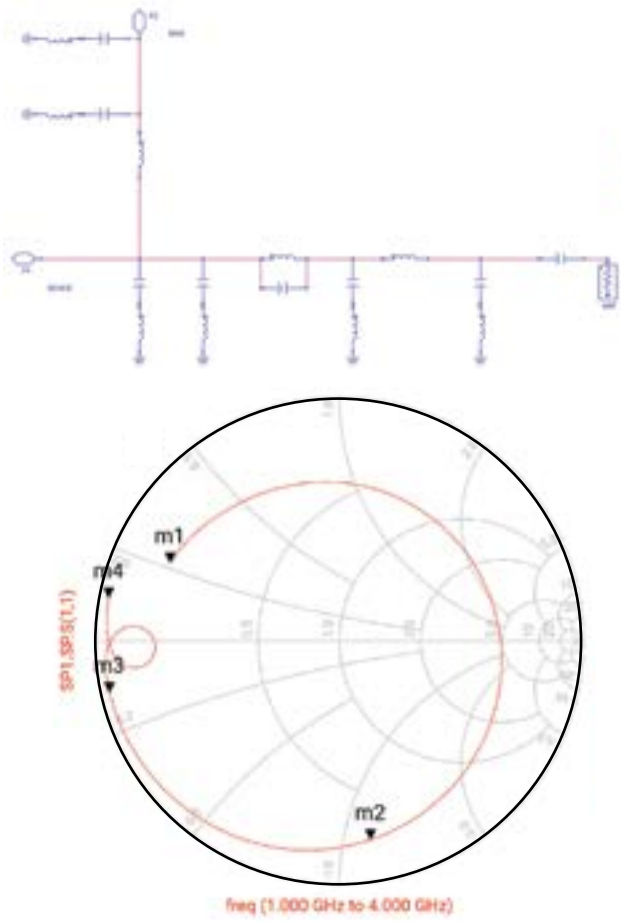
APPLYING AI FOR ADVANCED MODELING, ANALYSIS AND OPTIMIZATION

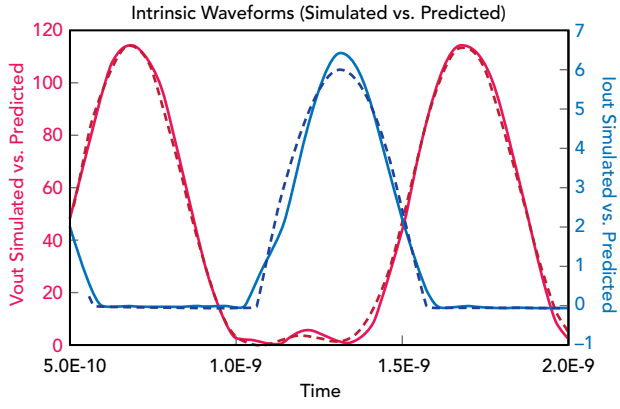
By capturing engineering processes, designers create well-defined interfaces that future AI tools can directly leverage. In AI-enabled workflows, agents are software programs that can operate autonomously, while skills are well-defined sets of instructions that agents or large language models (LLMs) can invoke to perform specific tasks. Diagramming the engineering workflow step by step gives engineers straightforward ways to use agents and skills productively. For example, an AI agent might extend the prior workflow to characterize all available transistors in a given technology and identify which devices perform best for a set of requirements. The agent or LLM might use a skill that describes how to set up and run a DCIV sweep and calculate the Class J load line. As AI tools evolve, these parts of the engineering process will converge, giving designers greater capabilities.

The waveform engineering orchestration in this ex-



▲ Fig. 9 **An ANN model of an inductor is validated and synthesized in a circuit via a Python script.**





Pout_simulated 47.5	DrainEff_simulated 74.9
Pout_predicted 47.5	DrainEff_predicted 74.6

Fig. 10 Output matching network tuned to Class J extrinsic targets using inductor surrogate models.

ample is only a small part of the larger PA design process. The prior workflow characterized the external impedances required at the device's output node to achieve textbook Class J waveforms. Its output might be a table of impedances at the fundamental and harmonic frequencies, which, when presented to the transistor, yield efficient device operation. The next step is to design a matching network that presents these impedances to the device. Rapidly evolving AI capabilities will likely streamline this process. For example, surrogate models backed by artificial neural networks (ANNs) could reduce or eliminate computationally intensive electromagnetic (EM) simulations and even enable on-the-fly synthesis of physical structures.

Surrogate modeling involves an upfront, sometimes resource-intensive training process that requires extensive characterization (for example, EM simulations). Once training is complete and validated, simulations and optimization using physical structures become ultra-fast, making surrogate models strong candidates for AI-enabled or fully autonomous design. Surrogate approaches have received considerable attention in recent years because they can synthesize non-traditional, QR-code-like physical structures with well-behaved frequency responses.[4] Recent research has even demonstrated the potential of such structures for harmonically matching a PA, as this example requires.[5] Note that surrogate modeling is not limited to pixelated structures; it can work for any parameterizable physical structure, even an inductor.

To illustrate, an ANN-based surrogate model was extracted for a set of inductors with varying geometries in a GaN MMIC process, as shown in ***Figure 9***. First, EM simulations are performed using Keysight RFPro for each unique set of inductor geometry parameters, such as width, spacing, number of turns and diameter. Next, a modified pi network with 11 components is fitted to the EM data. Then, for each circuit element in the pi network, a separate ANN is extracted using Keysight's proprietary ANN engine, available as a Python function in ADS.[6] The ANN produces Verilog-A functions that relate the physical parameters (w, s, N, D) to the circuit element values (capacitors, inductors and resistors). The result is a fast model that mirrors physical results and has input parameters that can be tuned or optimized.

The surrogate model was then used to quickly tune and optimize an output-matching network that presents the proper external impedances to the device, yielding near-ideal Class J operation, as shown in ***Figure 10***.

As more details of lower-level engineering processes are diagrammed and captured in Nexus Connect, routine tasks become in-

creasingly automated, allowing engineers using AI tools to focus on higher-level design. The path also leads to natural language interfaces that can augment or replace GUIs and scripts.

In summary, shifting from ad hoc simulations to executable, flowchart-based orchestration makes RF design workflows more repeatable, scalable and easier to share across teams. This approach enables engineers to capture complex design processes more effectively, exposes clear interfaces for automation and sets the stage for AI to accelerate design cycles while keeping engineers in control. ■

References

1. C. Pujol and M.Ozalas, "Reimagining Possibilities for Next-Gen Simulation in RF EDA," *Microwave Journal*, July 2025.
2. M. Ozalas,"A Synthesis-Based Approach to Quickly and Easily Design a Class E Amplifier," *Microwave Journal*, July 2015.
3. N. Khalid and A. Soury, "State-of-the-Art Load-Pull Simulation Methods for Next-Generation Power Amplifier Design," *2025 IEEE Wireless and Microwave Technology Conference (WAMICON)*, Cocoa Beach, Fla., USA, April 2025, pp. 1-4.
4. Y. Guo et al., "Dall-EM: Generative AI with Diffusion Models for New Design Space Discovery and Target-To-Electromagnetic Structure Synthesis," *2025 IEEE/MTT-S International Microwave Symposium - IMS 2025*, San Francisco, Calif., USA, 2025, pp. 926-929.
5. P. W. Shu, X. Zhou, T. Shama, L. Zhou and W. S. Chan, "Harmonic-Tuned Power Amplifier Using Artificial Intelligence-Assisted Topology Generation Algorithm," *IEEE Microwave and Wireless Technology Letters*, Vol. 35, No. 5, May 2025, pp. 561-564.
6. J. Xu, D. Gunyan, M. Iwamoto, A. Cognata and D. E. Root, "Measurement-Based Non-Quasi-Static Large-Signal FET Model Using Artificial Neural Networks," *IEEE MTT-S International Microwave Symposium Digest*, San Francisco, Calif., USA, 2006, pp. 469-472.

For more information on Keysight Technologies' products, applications, or services, please visit: www.keysight.com.
This information is subject to change without notice. © Keysight Technologies, 2026, Published in USA, July 22, 2026, 7126-1136.EN