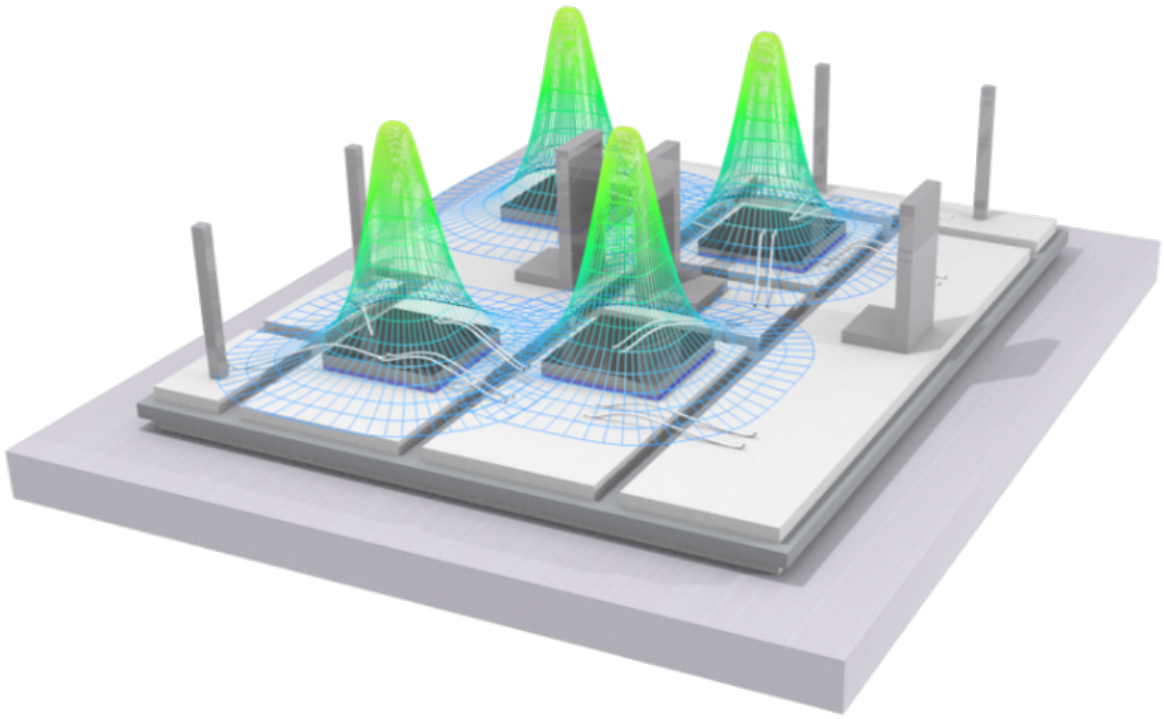
Mixed-Signal Computer-Aided Deisgn Laboratory

Energy-Efficient Electronics and Design Automation Laboratory

PowerSynth User Manual

April 24, 2019



Contents

1 Introduction

1.1 Executive Summary

PowerSynth currently performs multi-objective optimization to produce Pareto-front solutions to the proper placement of power semiconductor device die and the routing of metal traces on ceramic substrates. The tool accounts for temperature distributions and electrical parasitics as a function of the layout geometries that it considers. This tool has been hardware validated. Continued research on this project will further elaborate the capabilities by extending the work to greater fidelity in both the thermal and electrical domains.

Some of the major features of this tool include:

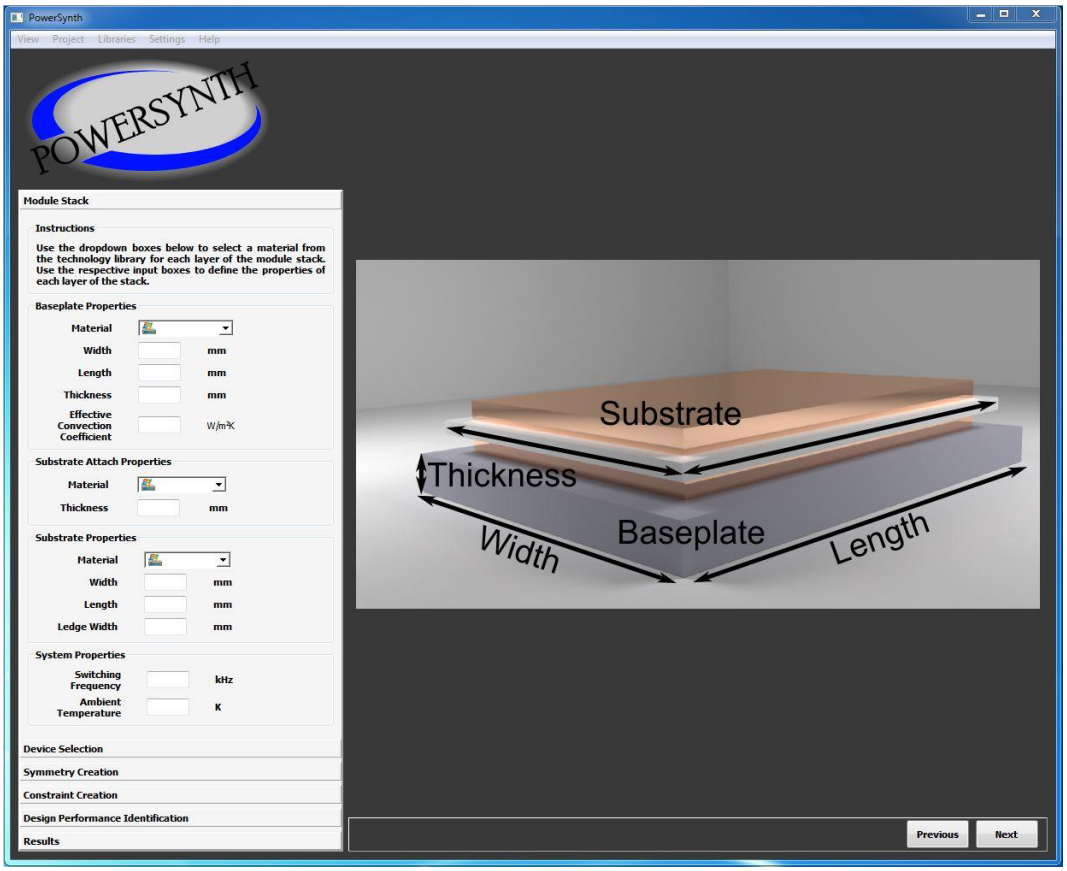
- Fast, accurate models for calculating electrical parasitics and thermal performance
- Multi-objective optimization for layout synthesis
- Able to synthesize and evaluate hundreds of layouts per minute
- User can select multiple performance metrics for optimization
- Built-in technology library including devices, substrates, attachment materials, bondwires, and more.
- Manufacturer Design Kit for incorporating packaging house design rules and tolerances
- Export designs to Ansys Q3D, FastHenry, or SolidWorks
- Post-layout optimization for trace corner filleting
- Easy to use GUI

1.2 Organization

After a brief introduction to PowerSynth, this document walks the user through various features of PowerSynth by creating a new project in a step-by-step presentation. Following the walkthrough, some additional information on the Technology Library, Process Design Rules, and Layout Editor are presented.

1.3 PowerSynth Overview

PowerSynth is an EDA tool with a friendly, graphical user interface for Multi-Chip Power Module (MCPM) layout synthesis. The tool is capable of importing a symbolic layout, which is an abstract layout representation of an MCPM, to automatically synthesize and generate multiple, real physical-layout solutions. This interface allows users to define technology libraries for power module materials, set up design constraints, and establish performance metrics for optimization. Additionally, export of optimal solutions is supported both graphically, on a 3D Solution Browser, and to modeling and analysis tools such as SolidWorks, Ansys Q3D, FastHenry, and Keysight EMPro.



The layout optimization of PowerSynth is based on fast thermal and electrical parasitic models developed specifically for this tool. The thermal model uses spatial superposition of temperature distributions to determine the temperature of devices in a module. The temperature and heat flux distribution of each device is characterized from a single run of an automated FEM simulation. In this simulation, a single die is placed in the center of the module atop and un-patterned substrate. The top surface of this device is treated as the source of power dissipation and a heat transfer coefficient is applied to the backside of the module heat spreader.

A compact thermal model (CTM) is then constructed from the results of this simulation. Thermal resistance values for the CTM are found by placing the characterized temperature distribution for each device in superposition with neighboring ones. Heat flux distributions obtained in the characterization step are used to account for the interaction among devices given their positions on the current trace layout. Once the CTM is constructed, linear algebra techniques similar to those found in circuit analysis are used to predict the temperatures of each device. The model can predict changes in temperature over variations in trace layout and device positioning with a maximum error of less than 10%.

The electrical parasitics in a module are approximated by closed-form solutions to micro-strip transmission line models. Since all power modules share the same basic geometry, the micro-strip model provides a reasonably accurate representation of a power module's trace parasitics. This is many times faster to execute when compared with finite element analysis. The electrical parasitic extraction works by breaking down a given layout into a lumped element network or graph with resistance and inductance values for each element or edge, approximated by micro-strip solutions. Capacitance from trace to baseplate (effective ground-plane) is evaluated at each lumped element node. These values can be used either directly in optimization or written into a netlist for simulation

purposes.

PowerSynth operates on an imported symbolic layout. This is a simple drawing of the topology of an initial module layout and is easy for a designer to create. A symbolic layout is comprised of three basic elements: lines, points, and rectangles. The line elements represent traces or bond wires. The point elements represent particular devices or leads, and the rectangle elements represent traces which span multiple traces vertically or horizontally in topological space. A designer chooses a set of performance metrics, in this case electrical or thermal, based on the symbolic layout. A multi-objective optimization problem is formulated based on the symbolic layout by allowing each layout line-element a variable width, which constitutes a set of design variables. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) is used to perform the optimization procedure.

After the optimization routine is run, a particular layout can be selected from a set of trade-off solutions for the MCPM design, allowing the designer easy access to the entire, viable design space. This system also allows a designer to quickly test many different layout solutions while maintaining layout quality. The fast electrical and thermal models both predict temperature and parasitic values accurately with respect to FEM tools.

1.3.1 Constraint-Aware Layout Engine

In this version, a constraint-aware layout engine has been integrated with PowerSynth. Now, there are two options for layout generation and optimization: 1) Using matrix-based layout engine (skip section 2.6), 2) Using constraint-aware layout engine (section 2.6). The significant improvements with the constraint-aware layout engine over the matrix-based one are:

- An interactive constraint input feature which is helpful for user to specify or modify design constraint values to have different layout structures.
- Four types of layout generation capability: minimum-sized layout, variable floorplan sized, fixed floorplan sized and fixed floorplan with fixed component locations, whereas the prvious one has only fixed floorplan sized layout generation capability.
- As the engine takes into account of all design constraints in the layout generation phase, it always generates 100% valid solutions, whereas the matrix-based engine generates 20-30% valid solutions due to design rule check(DRC)-failure.
- The matrix-based engine generates a non-smooth solution space and so gradient-based optimization algorithm cannot be used for optimization. But the updated layout engine can be used to test all different types of optimization algorithms. In this version two options are provided: NSGAII and non-guided randomization.
- Due to restricted input format (lines, points), the matrix-based engine is unable to process complex geometrical shapes in the layout. On the other hand, the updated layout engine treats each component as rectangle, so geometrical complexity is not a problem.
- The updated layout engine algorithms are more generic, scalable, and efficient than the matrix-based one. So, it can process broader range of layouts even considering heterogeneous components (e.g. gate drivers, EMI filters, sensors, etc.).
- As the updated layout engine is constraint-aware, different types of constraints can be declared : design constraints, reliability constraints, user-defined constraints. Generated solutions always satisfy all the given constraints.

In this beta-version, all of the features of the constraint-awrae layout engine are not integrated. Two layout engines are kept side-by-side, so that user can use both of the flows. The updated layout engine can take initial layout as a script describing all rectangle information that makes it more generic. However, in this version, the symbolic layout (lines, and points) format is considered as input format, which is automatically converted into rectangles in the constraint-aware layout engine for layout generation.

Methodology

From the user-defined initial input script, using corner stitch data structure (used in Magic VLSI tool), a collection of rectangular tiles are stored. Based on design constraints, constraint graphs (popular in VLSI floorplan compaction) are created from the corner-stitched plane. Two types of constraint graph are consdiered: horizontal constraint graph (HCG) and vertical constraint graph (VCG) for maintaining horizontal and vertical relationship among components.

- **Constraints**: Three types of design constraints are considered.
  1. **Dimension Constraints**: Here, minimum width along x-axis (Min Width), minimum width along y-axis and minimum enclosure are specified for each type of component. Currently, only trace, mos, diodes, and leads are considered.
  2. **Spacing Constraints**: In this table, minimum spacing values between every pair of components are declared.
  3. **Enclosure Constraints**: When a component is placed on top of another component, there may be some minimum enclosure value. So, this table has all possible minimum enclosure values.

- **Operating Modes**

  Based on the evaluation of the constraint graphs, there are four modes of operation (shown in Table 1).

Table 1: SUMMARY OF OPERATING MODES

Mode	Purpose	Evaluation Methodology
0	Minimum sized layout	Minimum constraint values
1	Variable floor-plan layouts	All weights are randomized with minimum constraints. No maximum constraints
2	Fixed floor-plan layouts	All weights are randomized with minimum constraints. Some have maximum constraints
3	Fixed floor-plan with fixed component locations	

  - **Minimum Size Layout**: This layout is generated using all minimum constraint values. So, this layout reflects maximum possible power density for a layout. As this is the minimum sized solution, it is electrically optimized but thermal performance is so poor.
  - **Variable Size Layout**: If this mode is selected, all constraint values are randomized and new layout solution is generated. User can generate arbitrary number of valid layout solutions with different floorplan size.
  - **Fixed Size Layout**: All edge weights are randomized within given area to generate arbitrary number of solutions. As floorplan size is always fixed there is less variation in this mode than the previous one.

– **Fixed Size with Fixed Locations**: This mode is useful for packaging purpose. If any component is to be always fixed at certain location throughout all solutions, this mode should be used. User can choose any absolute location at which any component needs to be fixed. In this mode, user can also generate arbitrary number of layouts.

All algorithms can be found in "WIPDA_2018.pdf" in the Publication folder.

2 Using PowerSynth

2.1 Installing PowerSynth

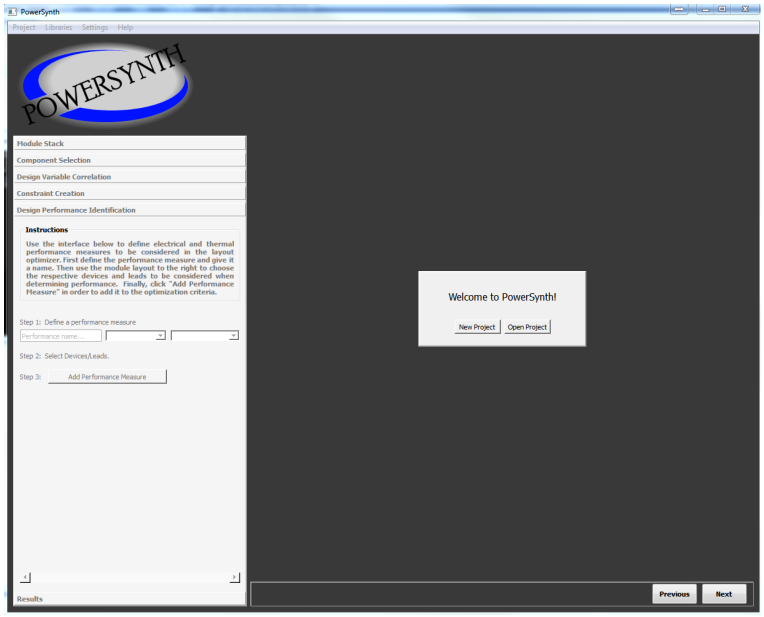
1. Run the PowerSynth install wizard
2. Run PowerSynth.exe

Note: It is STRONGLY RECOMMENDED that you install PowerSynth in the C drive (the default installation directory) and NOT in Program Files. Installing PowerSynth in Program Files can result in various issues relating to administrator permissions.

2.2 Creating a New Project in PowerSynth

2.2.1 User Interface

The welcome screen is shown on the right. The navigation pane on the left is used to develop a project once a new project is created or an existing project is opened. Before we click "New Project", we need to define the layout template that will be imported.



2.2.2 Symbolic Layout File

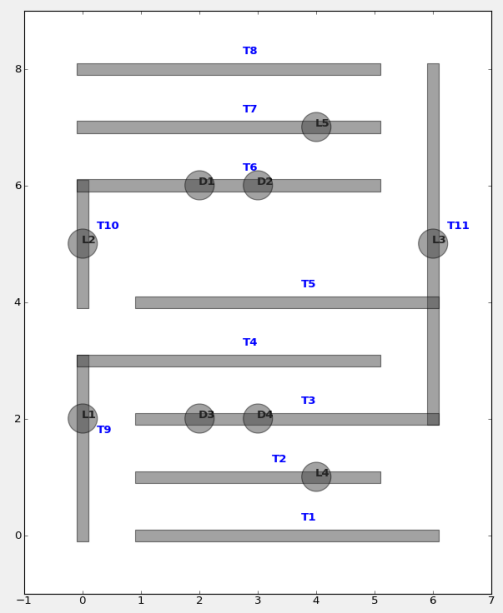
A symbolic layout is a simple stick diagram used to represent a simple power module's abstract layout. In the previous version, PowerSynth relies on InkScape to create a SVG file. This SVG file represents a stick diagram which is then imported into PowerSynth for a new project. In this version, SVG feature has been removed due to some unstable functionalities. Instead, the user can provide a script (text file) to define the stick diagram. Each point on this stick diagram represents either a device or a lead, while each line represents a trace. Bondwires information will need to be provided later in the setup (see Assigning Virtual Wires Connections). This script must be created before creating a new project in PowerSynth. Open a text file and define the layout template using lines and points with the syntax as shown below. Lines denote traces, while points denote devices and leads. Below is a sample layout script. It is of the format:

Name Type (x1,y1) (x2,y2)

Note: The element name (first column) must be four characters long. For a point, (x1, y1) is the coordinate of the point. For a line, (x1, y1) is one of the end points. (x2, y2) is the other endpoint. Save this file in either .psc or .txt format.

The example script for the symbolic layout is shown below:

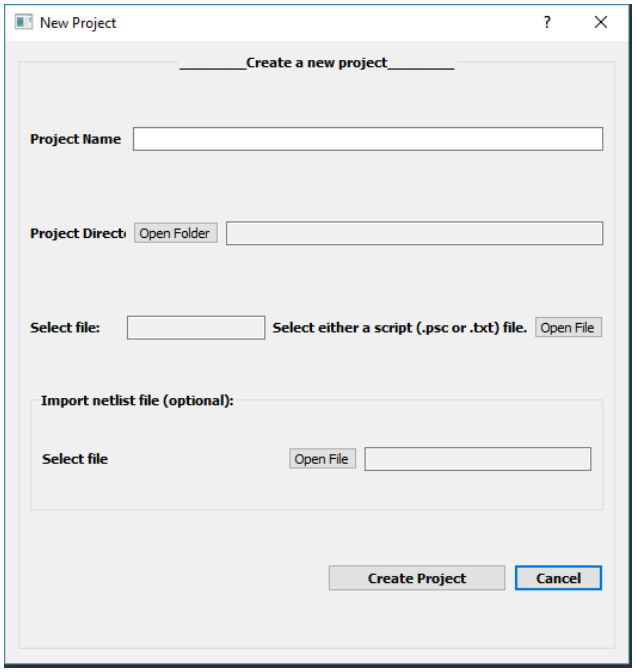
```
T1 l (1,0) (6,0)
T2 l (1,1) (5,1)
T3 l (1,2) (6,2)
T4 l (0,3) (5,3)
T5 l (1,4) (6,4)
T6 l (0,6) (5,6)
T7 l (0,7) (5,7)
T8 l (0,8) (5,8)
T9 l (0,0) (0,3)
T10 l (0,4) (0,6)
T11 l (6,2) (6,8)
D3 p (2,2)
D4 p (3,2)
D1 p (2,6)
D2 p (3,6)
L1 p (0,2)
L2 p (0,5)
L3 p (6,5)
L4 p (4,1)
L5 p (4,7)
```



Note that these coordinate values are not real coordinates and thus have no dimensional units. These coordinates are used to find the relative position of each line and point to each other. Once this file is imported to the software, an automatic normalization algorithm will run to count the number of rows and columns. These coordinates are then normalized between zero and the maximum number of rows/columns. The corresponding symbolic layout for the abovementioned script is represented on the left. Once the project is loaded, the layout can still be modified using the Layout editor under **Project -> Open Layout Editor**

2.2.3 Create a New Project

Selecting the "New Project" button on the welcome window brings up the dialog for a new project as shown on the right. First, give the project a name. Then, select the folder where it will be saved. Import the symbolic layout file as shown above using "select file". Select the directory where you saved the text file. Finally, click "Create Project".



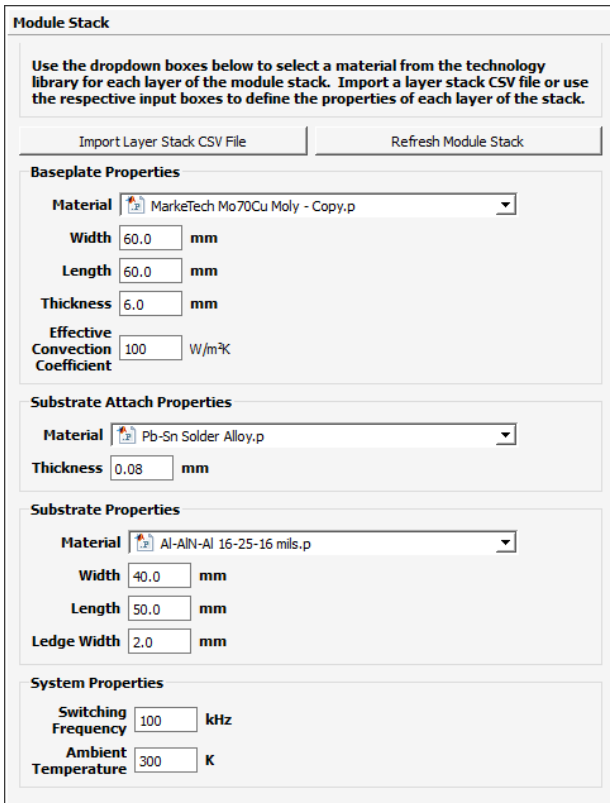
Optional: The user can also specify a netlist file that have same circuit topology with the symbolic layout. This can be later used to map the net names with the symbolic layout object. A sample netlist format for a 4 switches half bridge is shown below:

```
*Sample Netlist file:*
M1 DC_plus G_High Out Out NMOS
M2 DC_plus G_High Out Out NMOS
M3 Out G_Low DC_neg DC_neg NMOS
M4 Out G_Low DC_neg DC_neg NMOS
.model NMOS NMOS
.model PMOS PMOS
.end
```

2.3 Defining the Module and Components

2.3.1 Module Stack

The module stack tab on the left allows the user to input module stack information for the design. This includes dimension and material information of the substrate, substrate attach and baseplate. Operating conditions such as ambient temperature, effective convection coefficient of baseplate and switching frequency are also required for thermal and electrical evaluations. The user can update the tech files (**.p) using * Libraries->Tech lib**.



Another way to import this module stack is through a csv layer stack file. The description for this file is explained in the next section.

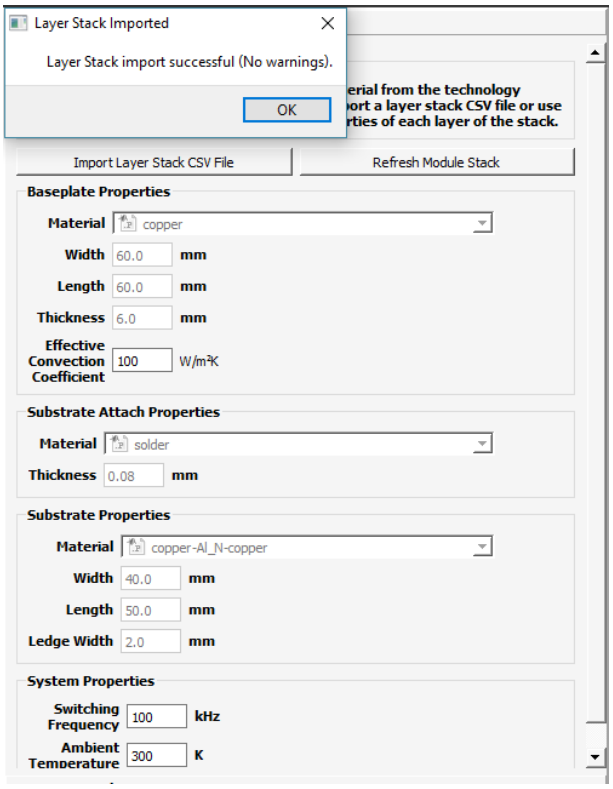
2.3.2 Layer Stack File

Abbrev	LayerID	Name	Layer Position	Width	Length	Thickness	Material
B:	1	B1	1	45	45	6	copper
SA:	2	SA1	2	40	40	0.08	solder
M:	3	M1	3	38	38	0.41	copper
D:	4	D1	4	40	40	0.64	Al_N
I:	5	I1	5	28	28	0.41	copper
C:	5	C1	6	2.5	4	0.2	SiC

The Table above shows the module stack structure. The user can specified this in an excel (.csv) and load it using the **Import Layer Stack File** button in the UI.

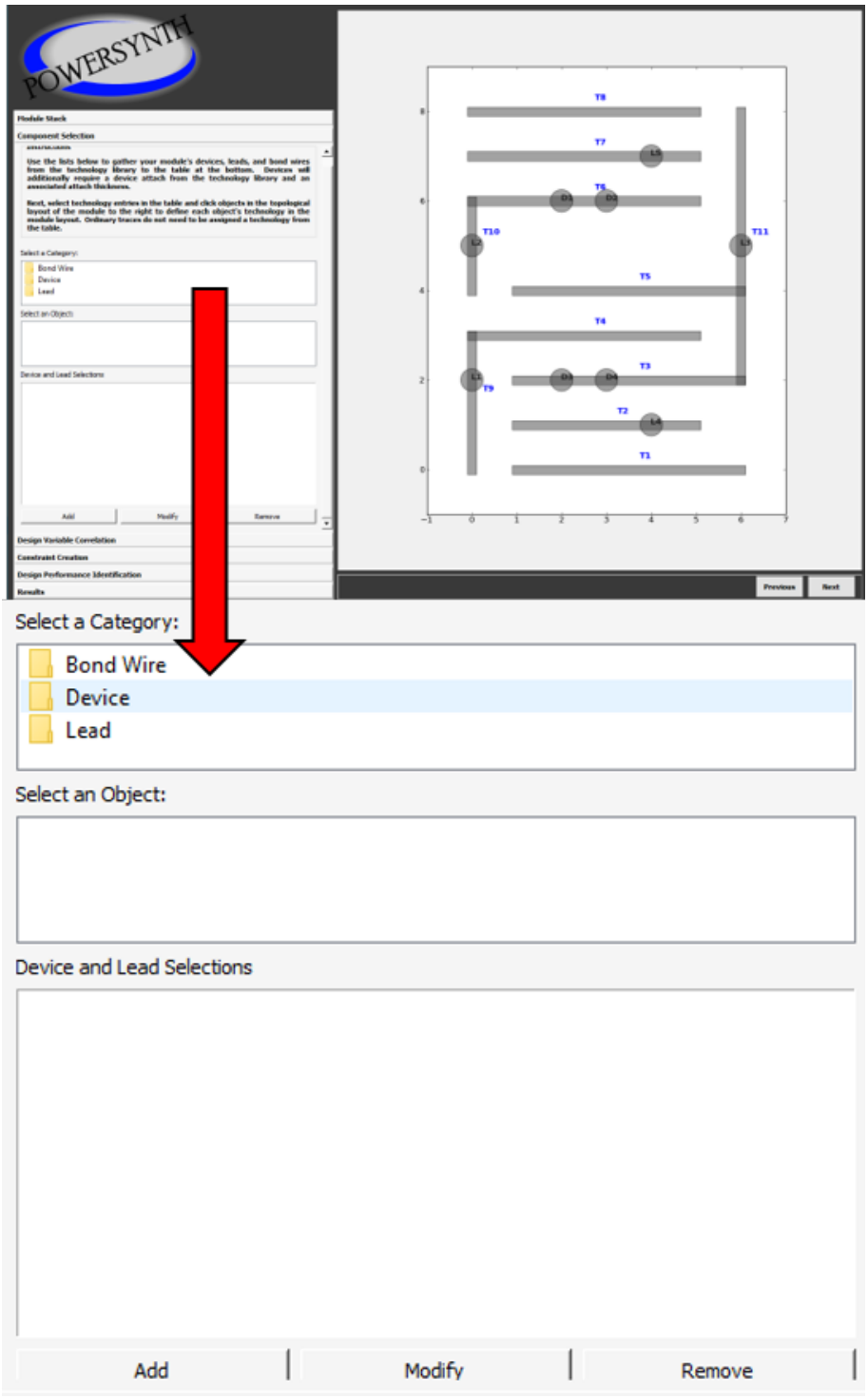
Some Abbreviation: B: Baseplate, SA: Substrate Attach, M: Metal, D: Dielectric, I: Interconnect (Metal), C: Component (this is optional for this version).

Below shows the UI after successfully importing the layer stack. To go back to the normal mode, press the **Refresh Module Stack** button.



2.3.3 Component Selection

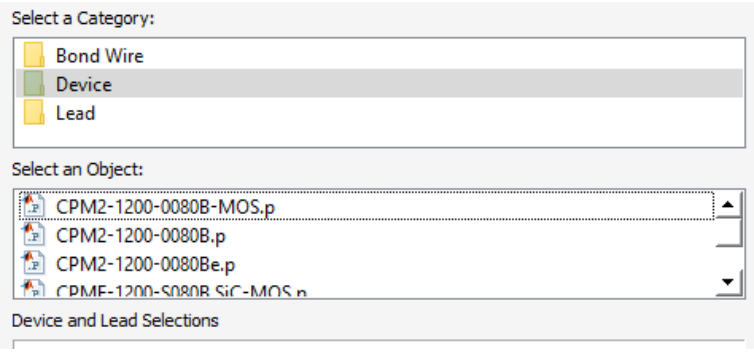
The symbolic layout is shown on the right side of the interface. This allows users to insert component technology files. These files contain information about the bondwires, leads, and devices the user wishes to include. Notice that the names from the layout script for each traces and Devices are shown on the right-hand side. (Symbolic Layout)



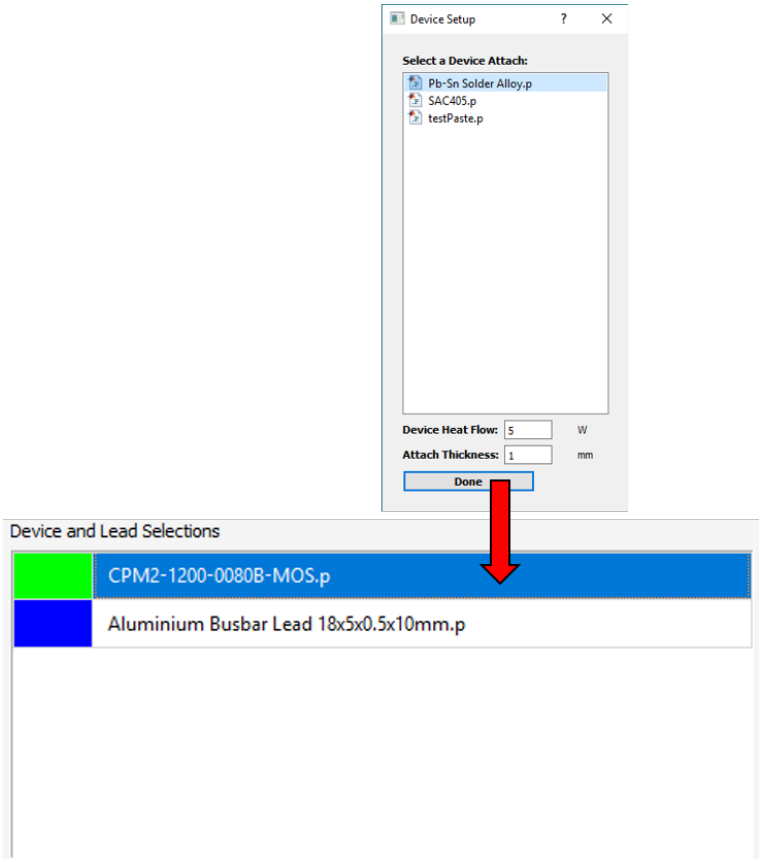
The user can assign devices and leads to the circles on the symbolic layout. The unassigned lines will represent copper or aluminum traces (depending on substrate selection). Wires bond will be treated as virtual connections. However, it is recommended to finish the devices and leads selection first before assigning virtual wire bonds to ensure correct connection.

2.3.4 Device and Lead Selection

Click on "Device" or "Lead" from the Category list, a list of objects under object selection will be listed out. This can be specified using the TechLib (See Technology Editor section)



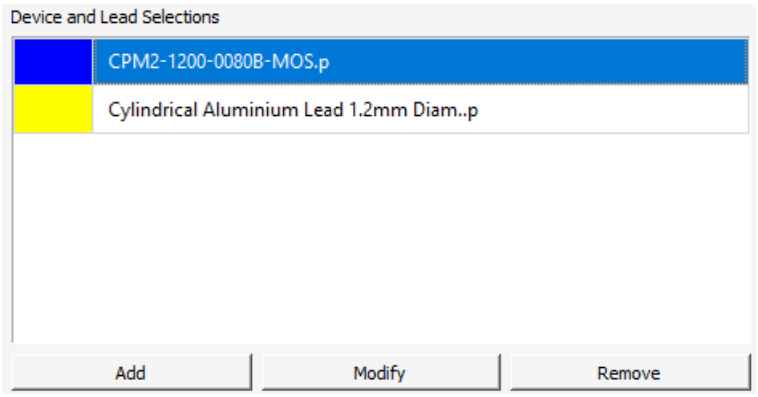
Once an object is selected, press the "Add" button. In case of device, this will pull up a dialog on the right to define the solder type (tech lib), solder thickness and average power of the device. This information will be used for thermal optimization. Click Done to apply this for the device. This will update the device and lead selection table. The user can do the same for the lead selection to fill the Device and Lead Selection table on the right.

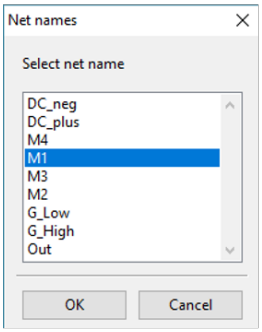


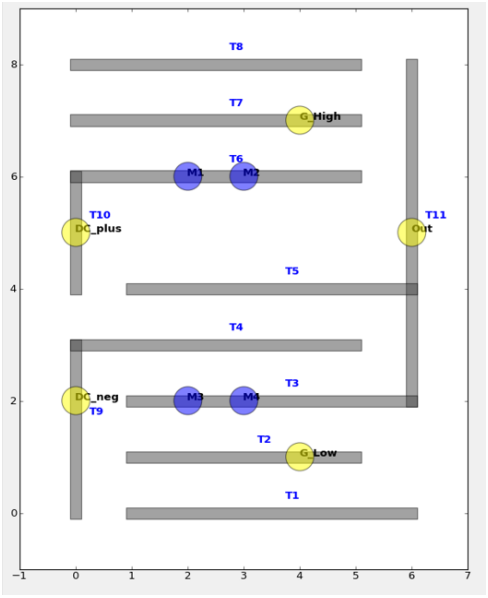
2.3.5 Assigning Devices and Leads

First, select a device or lead object from the table.

Assigining these selections on the Symbolic Layout by clicking on the circles. To deselect, click on the selected circle one more time. The circles will be colored based on the random generated color on the Device-Lead Table. In case the user imported a netlist (Create new project section), a small dialog will be trigged to select the corresponded net name.

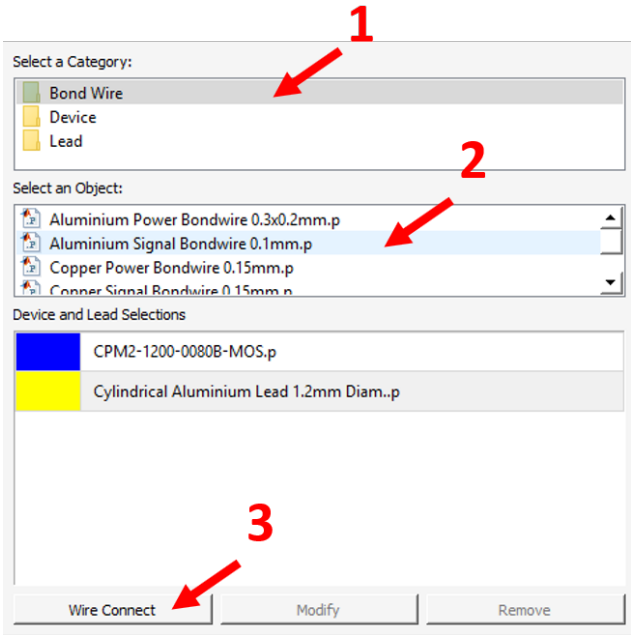






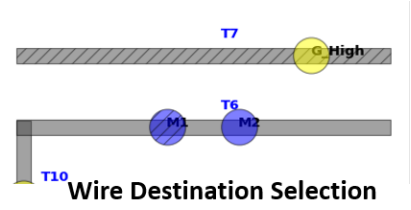
2.3.6 Assigning Virtual Wire Connections

Once all of the devices and leads are setup like above, the next step is to setup the virtual wires connections. To do this, first click on "Bond Wire" from the categories selection list. Then, select a bondwire object. The "Add" button will change its text to "Wire Connect" as seen on the right:

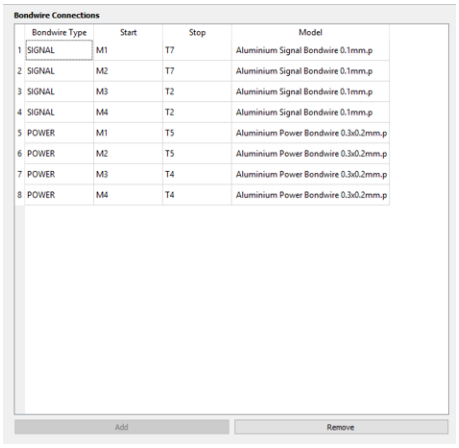


A bond wire Setup dialog will open as seen on the right. The symbolic layout is now in wire selection mode. The user will need to select 2 traces or a trace and a device (device to device bondwire is not supported yet) for wire connections. For a MOSFET, it is required to have signal and power wires for signal and source sides. To select 2 objects, simply click on the symbolic layout again. The selected object will be hatched as shown below.

The **Add** buton on the dialog will be enabled if 2 objects are selected (click again if you want to deselect an object). Click this to add the wire connection to the table. Doing so, we can have a complete table (on the left). Close the dialog, the table can be saved and edited any time.



Wire Destination Selection

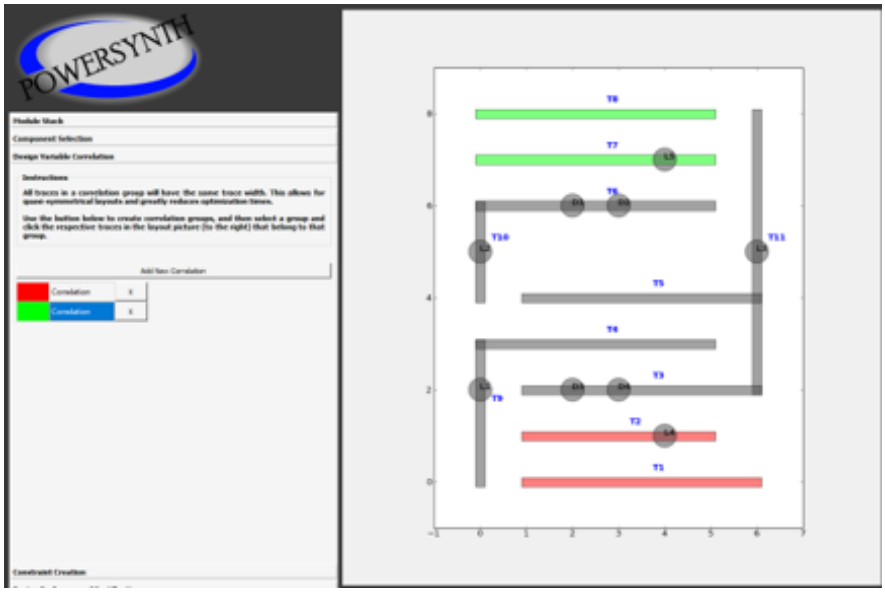


Full Virtual Connection Table

2.4 Defining Correlations and Constraints

2.4.1 Design Variable Correlation

If symmetrical traces or device positions are desired in the design, correlation between devices or traces can be assigned. This will ensure that selected traces will have the same width (length) or that selected devices will have same position. Click "Add New Correlation" and a new row will appear in the table. Click on a row then click on traces or devices to specify a correlation.



2.4.2 Constraint Creation

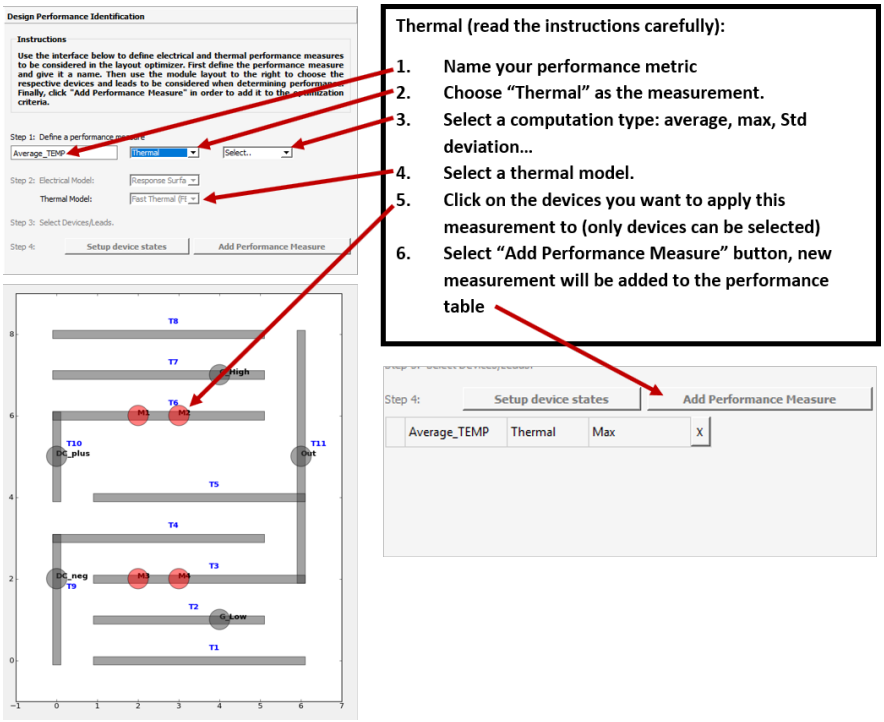
Here, the user can click on each line (trace) to specify a min and max or fixed width for a trace. If there are symmetry constraints, these will be applied to all symmetric traces.

2.5 Design Performance Identification

In this step, the user can choose the performance metrics that are to be optimized. The next two sections below show the process for assigning performance metrics. Once the performance metrics are selected, the user can go to the next tab to run the design optimization. Currently, PowerSynth supports thermal and electrical measurement of a power module.

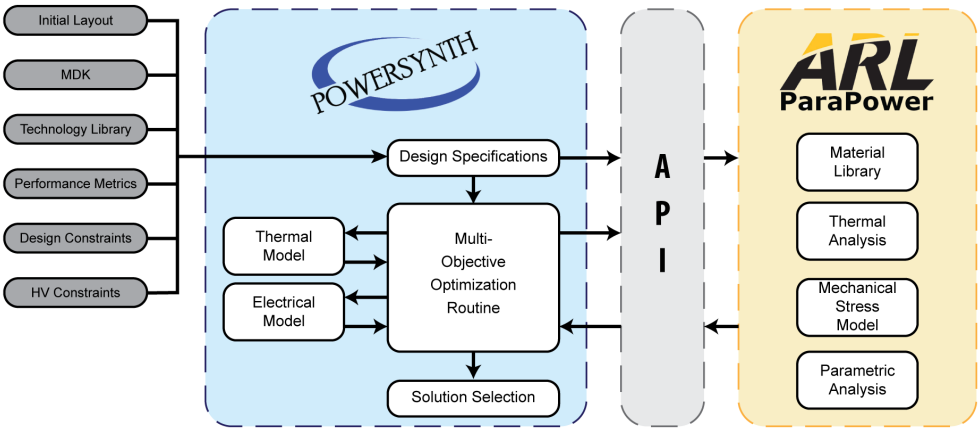
2.5.1 Thermal Measurement

On the navigation tool, a user can define a performance measure. There are 2 available thermal models in PowerSynth: Rectangle Flux (analytical) and Fast Thermal (Characterized by FEM simulation). In case of thermal measurement, follow the steps below:



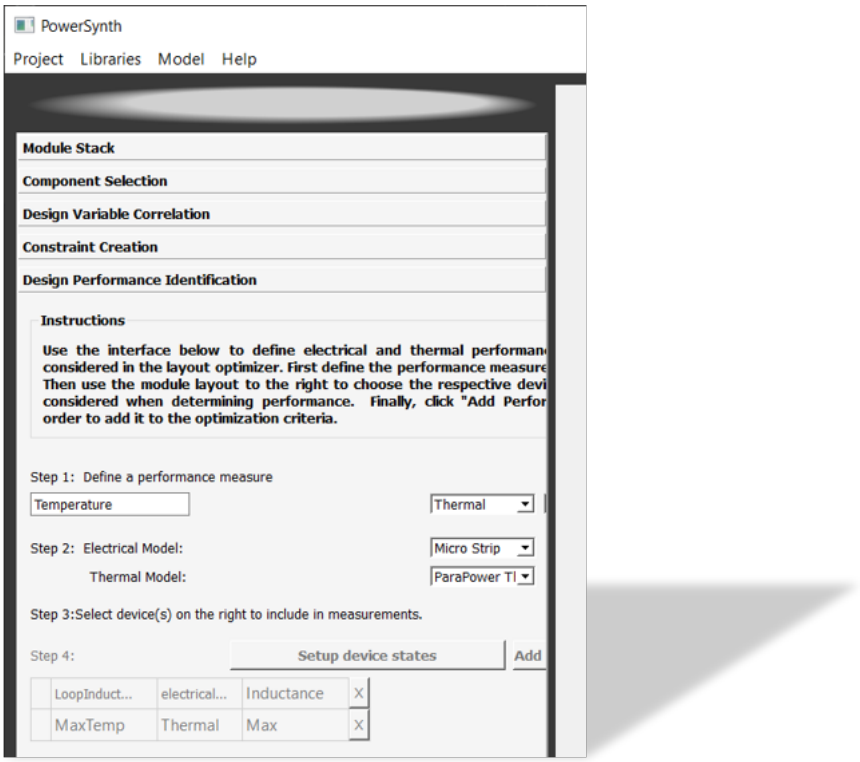
2.5.2 ARL ParaPower Thermal Analysis

As part of a collaborative project with the US Army Research Lab (ARL), an API has been developed to utilize their tool, ParaPower, within the PowerSynth optimization routine. ParaPower provides quick steady-state and transient thermo-mechanical analysis capabilities that can be applied toward the electro-thermo-mechanical co-design of power modules. A brief overview of how the ParaPower API used by PowerSynth is shown below. Additionally, more information regarding this work can be found in ECCE article mentioned in 4.



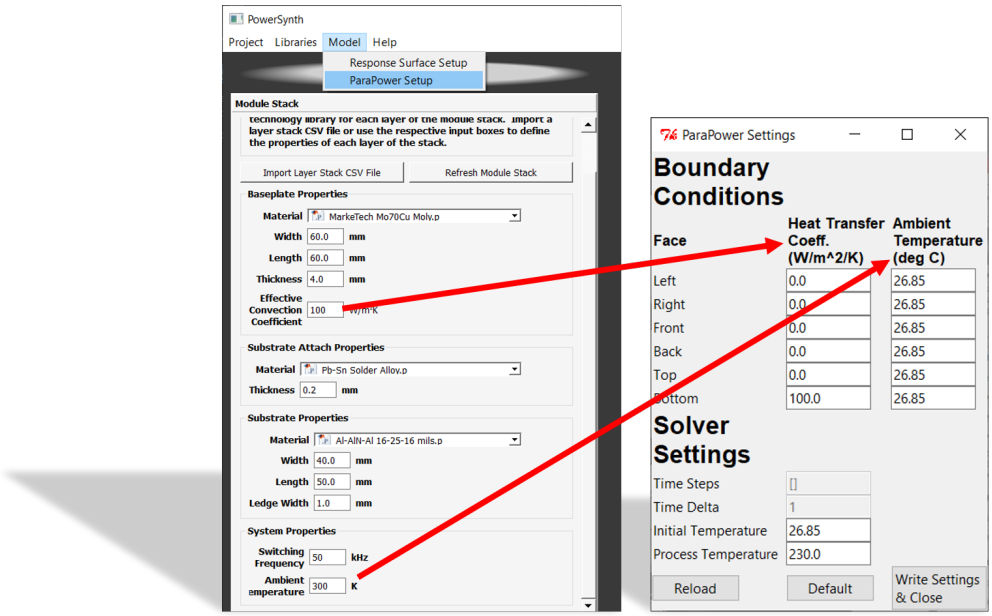
While access to the full analysis capabilities of ParaPower are available through the developer version of this API, for this release of PowerSynth (v1.4) only steady-state thermal analysis is supported. One other caveat is that this is meant to work with the legacy "symbolic layout" power module representation used in PowerSynth and not the newer, layer-stack representation.

To use ParaPower steady-state thermal analysis as a design performance criterion, similar steps are performed as in section 2.5.1. The main difference being that when choosing the thermal model, the "ParaPower Thermal Model" option should be selected as shown below:



ParaPower offers additional boundary condition settings beyond what this release of PowerSynth considers using the built-in thermal models. The built-in PowerSynth modules only consider temperature and heat transfer from the module backside. Whereas, with ParaPower, these conditions may be specified for all six faces of a rectangular prism enclosing the power module design.

ParaPower analysis boundary conditions and solver settings can be accessed at any point prior to optimization in PowerSynth by choosing "ParaPower Setup" from the "Model" drop-down on the main menu as shown below. Values are automatically populated from the information provided in the "Module Stack" settings pane. Note that these are automatically chosen to mimic the boundary conditions of the other PowerSynth thermal models. If boundary temperature and heat transfer values for any of the other five faces are to be specified, they can be entered manually and then the saved by choosing "Write Settings & Close."

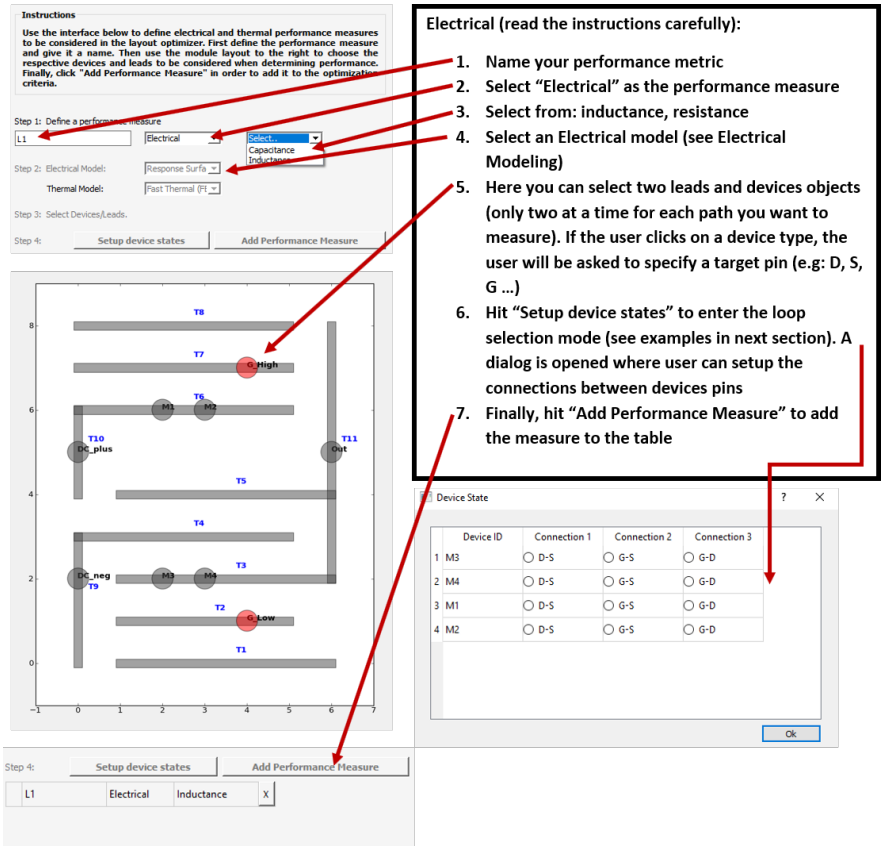


After ParaPower setup has been completed, along with choosing the ParaPower Thermal design performance criterion, PowerSynth optimization can be executed as normal with the maximum temperature results being added to the solution Pareto frontier.

2.5.3 Electrical Measurement

Parasitic capacitance, resistance, and inductance can each be specified as design performance metrics to be optimized. The next few sections will detail the steps necessary to define them and build response surface models for quick evaluation.

2.5.4 Inductance and Resistance Measurement

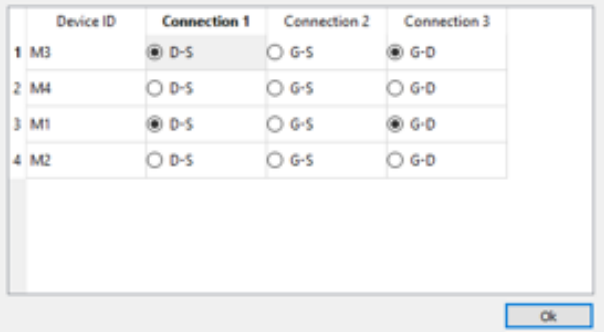


2.5.5 Device Connection Setup (Example)

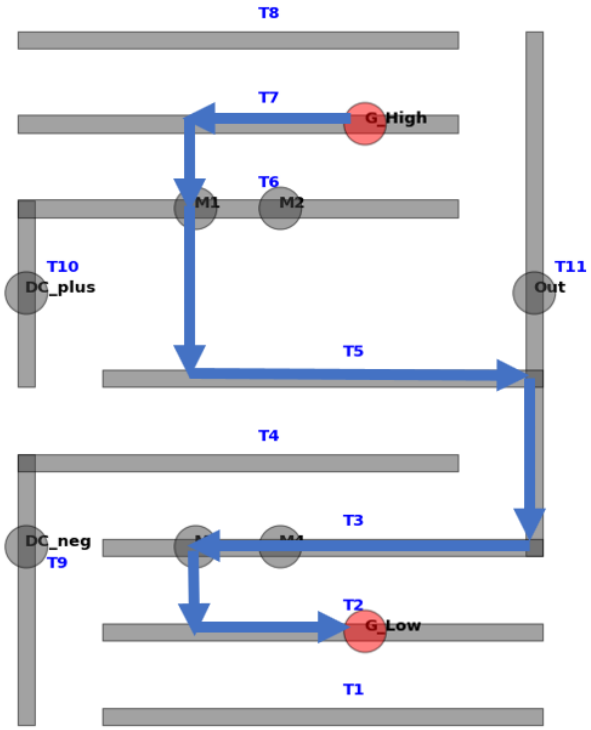
Although on the symbolic layout devices are represent as nodes (circles), in a real power module layout, these devices have real dimension and several connection pads. By default, there are no connections between these pads (usually a capacitance if this is a MOSFET). To give the user the flexibility to select different loops, the setup device connection feature is used to select virtual connections between any two pins of a given devices. For example, from the same electrical measurement above, we can setup 2 different loops.

Loop 1:

Setup the device connections as follows:



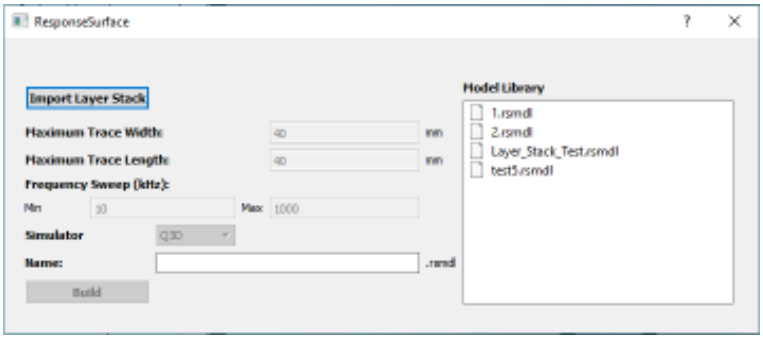
We now have the following loop:



Name this loop as `L1` and add this to the table.

2.5.6 Electrical Modeling

Under the electrical model selection on the UI, the analytical microstrip option can provide quick estimation of the inductance and resistance results. However, there are several drawbacks for this model such as not-frequency dependent, and usually less accurate for big traces. To overcome this problem, a response surface option is also provided. This model can be characterized using FastHenry or Q3D. To open the model creator, the user can go to: **Model -> Response Surface Setup**. A dialog as seen below is opened:



1. **Import Layer Stack:** this is explained in the Module Stack section below, this defines the baseplate and substrate dimensions. Different baseplate structures can affect the inductance calculation.

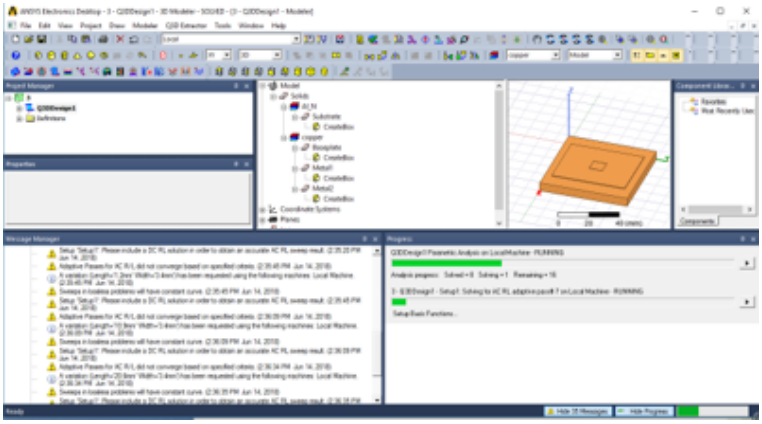
2. **Defining Maximum Trace Width and Length:** minimum trace width and length are based on minimum trace width in the Design Rule Check from PowerSynth (See Process Design Rules Editor). Define the maximum length and width here to characterize the model

3. **Defining Frequency sweep:** Specify the minimum and maximum frequency in KHz. The maximum frequency will define the minimum mesh in the characterization.

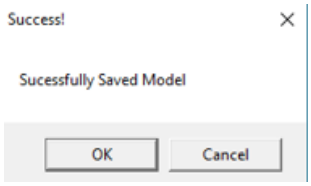
4. **Simulator:** 2 simulators are available now. These include ANSYS Q3D and FastHenry, the default ANSYS directory is: `C:\Program Files\AnsysEM\AnsysEM18.2` (Versions up to 18.2 are supported). Another option is FastHenry (recommended), this is an open source parasitic extraction tool from MIT. The exe file for FastHenry is included in the package.

5. **Model name:** Model name can be specified by user. This will be later added to the model library on the right.

6. **Build:** Once everything is setup, hit "build" button. Depending on the Simulator choice, and maximum frequency, the model will be built within 15~30 minutes.

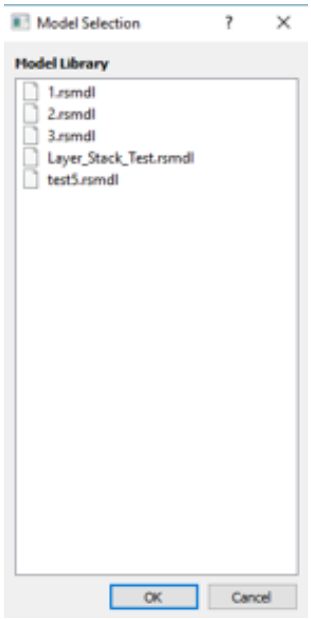


The simulation is automatically run in Q3D to characterize the model. If FastHenry simulation is selected this will be run in the back-end (Command Prompt) of the tool.

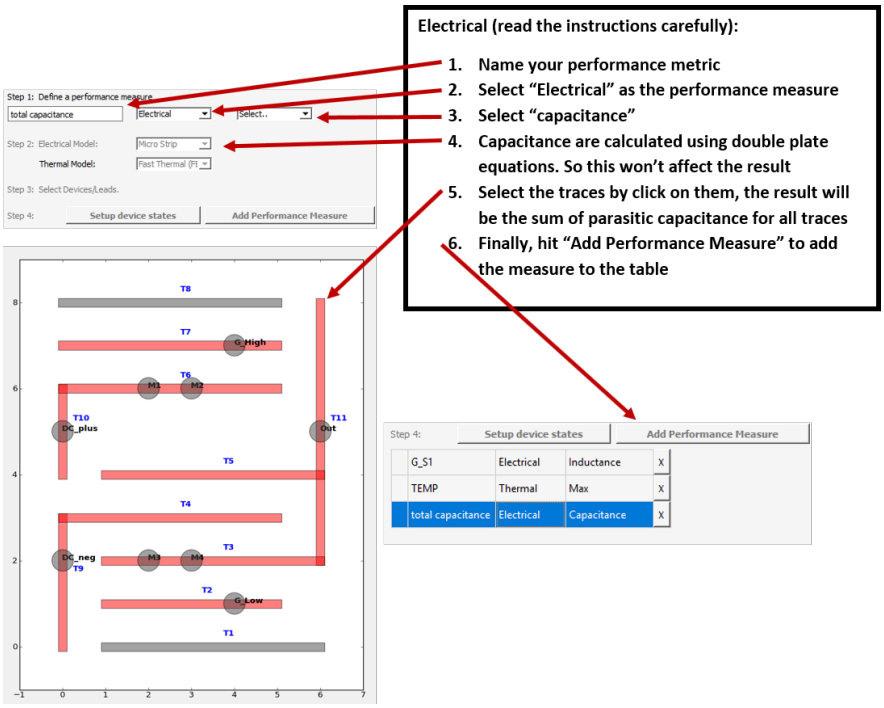


A dialog shows up when the simulations are done. Click Ok to go back to PowerSynth UI.

If the model is selected in the design performance selection, a dialog will be opened. The user can select the models from the list:



2.5.7 Capacitance



2.6 Constraint-Aware Layout Engine (Beta-Version)

To use this layout engine, the user must follow the steps up to 2.3.6. Then constraint-aware layout engine button will be enabled. Upon clicking on the button (marked with an arrow in Fig. 1) named **"Constraint-Aware Layout Engine"** the updated layout engine dialog will appear (shown in Fig. 2).

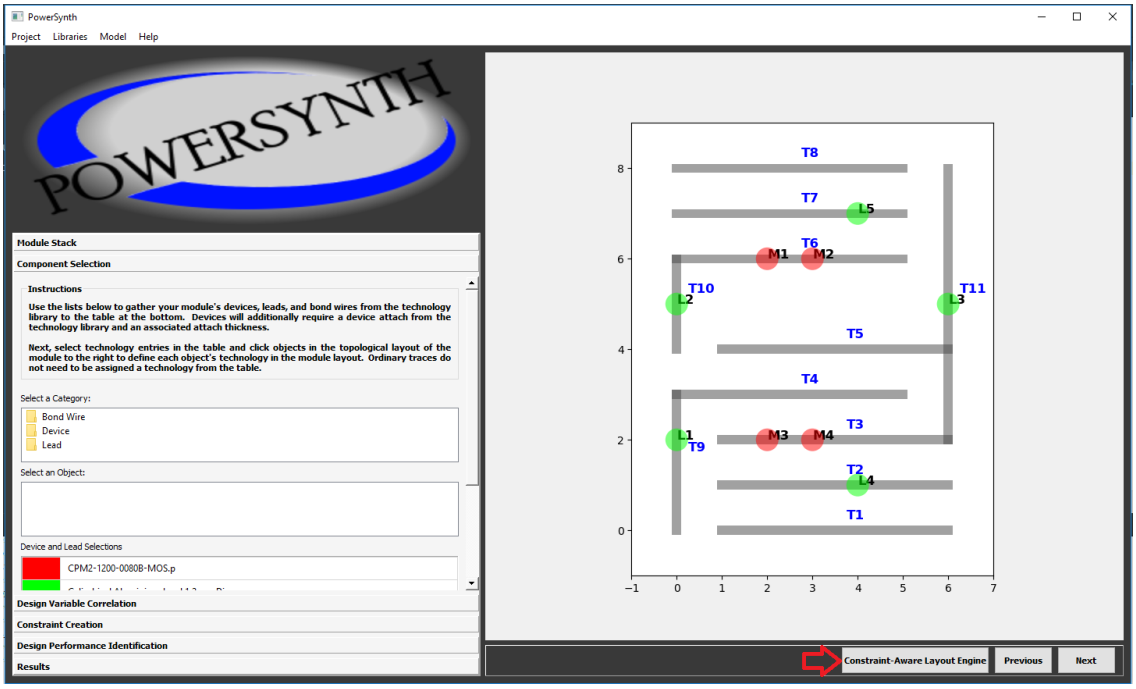


Figure 1: Entering into constraint-aware layout engine

2.6.1 Constraint-Aware Layout Engine Dialog

The new window has four parts.

1. **Generated Layout**: In this part, layout solution will be appeared.
2. **Input (Initial) Layout**: Here, the initial layout that is converted from symbolic layout will appear. This is just rectangular representation of the symbolic lines and points objects.
3. **Layout Selection**: This is the solution browser. Here, all solutions will be appeared as dots. Upon click on each dot corresponding layout will be shown in top left part (marked as 1 in the Fig. 2).
4. **Control Features**: In this part, all information are shown and necessary input information are taken. Here, floorplan size of each layout, controlling features of the layout engine, different operating mode selection option are displayed.

2.6.2 Assign Constraints

At the starting of the constraint-aware layout engine, the constraints must be assigned. To assign constraints, the **"Assign Constraints"** button should be clicked and 'Constraints setup' window

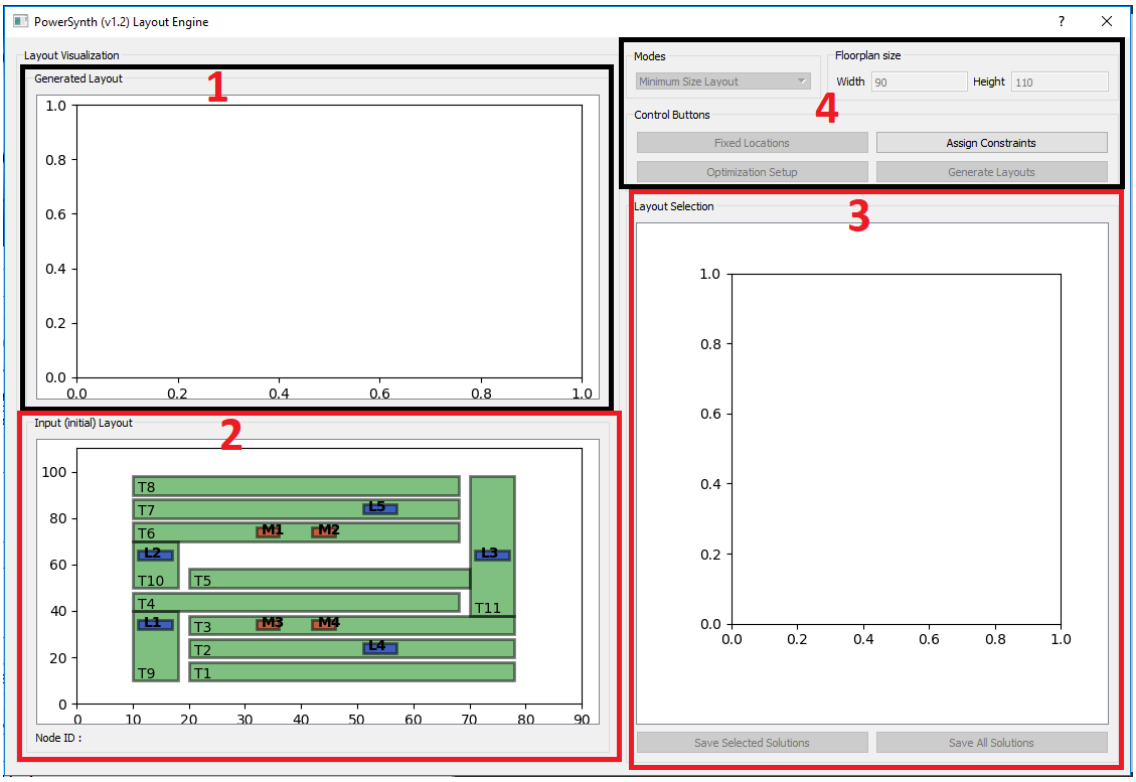


Figure 2: New Layout Engine Window

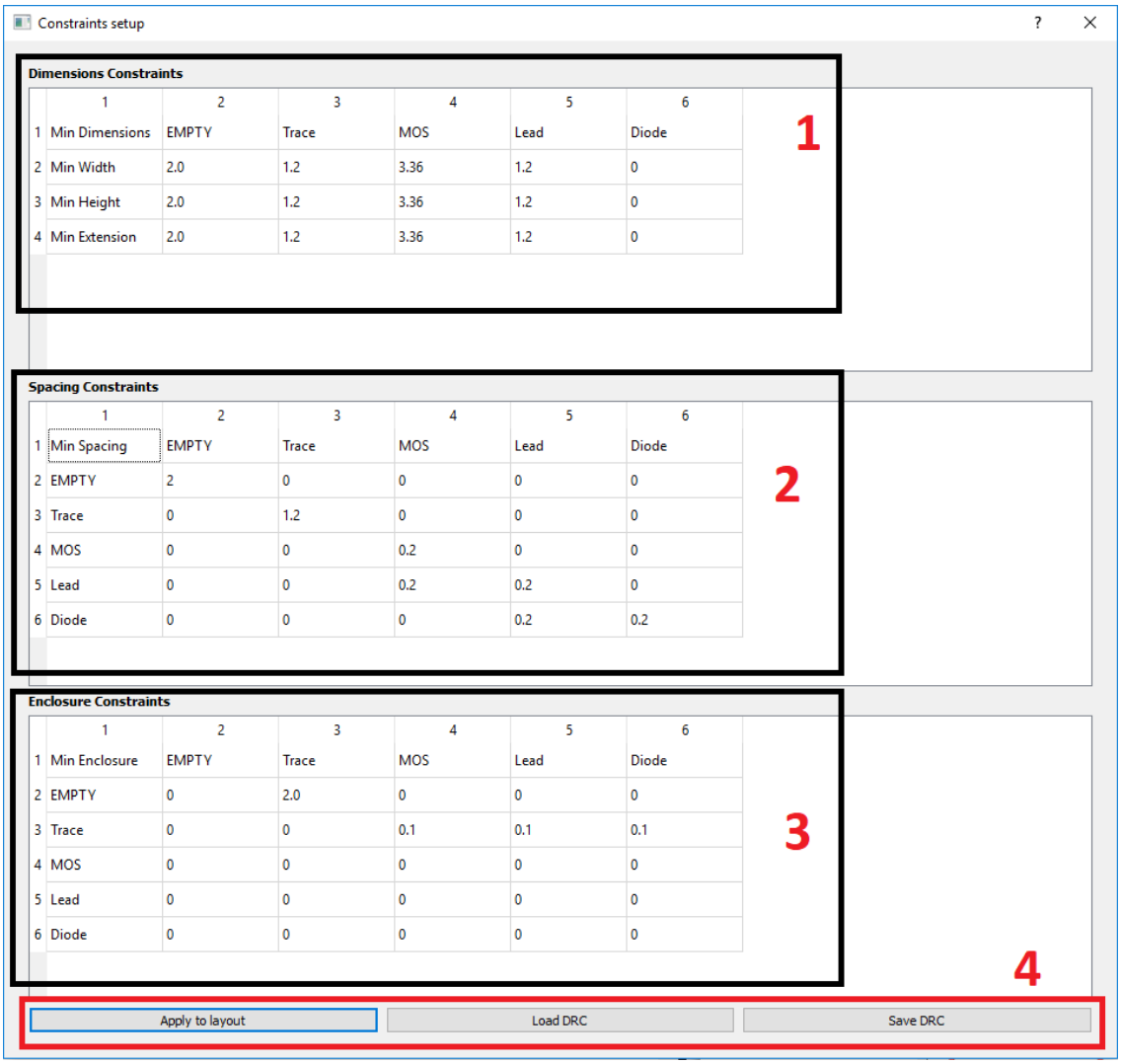
will appear (shown in Fig. 3). There are four parts in the constraint setup: 1) Dimension Constraints, 2) Spacing Constraints, 3) Enclosure Constraints, and 4) Setup buttons. All constraints are automatically updated according to PowerSynth technology library. But the values can be modified by the user. Any modification can be performed in this table before clicking **"Apply to Layout"**.
Setup Buttons: After setting up the constraint values, to apply the constraints to the layout solutions **"Apply to layout"** button must be clicked. User can also load a constraint table from a csv file (in the same format shown in the Fig. 3), by using **"Load DRC"** button. **"Save DRC"** button can be used to save the modified constraint table in csv format so that later it can be used.

Assign constraints is the first and the most important part for the layout engine. Without performing this step, layout solution generation is impossible.

2.6.3 Modes

To select the mode of operation, the drop down list under **"Modes"** can be used. The list is shown in Fig. 4. According to choice of the user the proper **"Mode"** should be selected, before generating layout.

- **Minimum Size Layout**: In this mode, only one layout will be generated. So, in the **"Optimization Setup"** option, only "Design Performance List" should be populated.
- **Variable Size Layout**: In this mode, user can generate arbitrary number of layouts. So, in the **"Optimization Setup"** option, along with "Design Performance List", **"Number of Layouts"** and **"Seed"** need to be provided.



	1	2	3	4	5	6
1	Min Dimensions	EMPTY	Trace	MOS	Lead	Diode
2	Min Width	2.0	1.2	3.36	1.2	0
3	Min Height	2.0	1.2	3.36	1.2	0
4	Min Extension	2.0	1.2	3.36	1.2	0

	1	2	3	4	5	6
1	Min Spacing	EMPTY	Trace	MOS	Lead	Diode
2	EMPTY	2	0	0	0	0
3	Trace	0	1.2	0	0	0
4	MOS	0	0	0.2	0	0
5	Lead	0	0	0.2	0.2	0
6	Diode	0	0	0	0.2	0.2

	1	2	3	4	5	6
1	Min Enclosure	EMPTY	Trace	MOS	Lead	Diode
2	EMPTY	0	2.0	0	0	0
3	Trace	0	0	0.1	0.1	0.1
4	MOS	0	0	0	0	0
5	Lead	0	0	0	0	0
6	Diode	0	0	0	0	0

Figure 3: Constraint Table

- **Fixed Size Layout**: To have fixed floorplan sized layout solutions, user should enter **"Width"** and **"Height"** corresponds to floorplan size of the generated layouts. In this mode, user can choose **"Optimization Algorithm"**. If "NSGAII" is selected **"Number of Generations"** needs to be given, otherwise **"Number of Layouts"** needs to be provided. Also, **"Seed"** is a must.

- **Fixed Size with Fixed Locations**: In this mode,**"Width"**, **"Height"** must be given. Besides, one additional requirement is the **"Fixed Locations"**.This button is used for allocating fixed coordinates to the desired nodes. On this mode, the input layout is updated with clickable dots on possible nodes of the layout as shown in Fig. 5 with marked as "1". To fix a node at certain coordinate, user needs to click on the node and the **"Node ID"** will appear at the bottom. Then, **"Fixed Locations"** button needs to be clicked to find the desired "Node ID". Upon selecting the "Node ID" from the drop down list, the valid x and y coordinate range will appear for that node. User needs to choose a value within the range and then **"Add"** button needs to be clicked to store the location. In the similar way, as much nodes as user want can

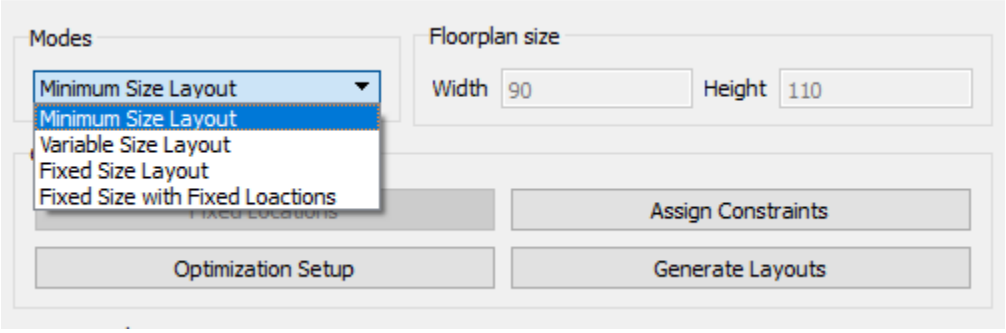


Figure 4: Modes

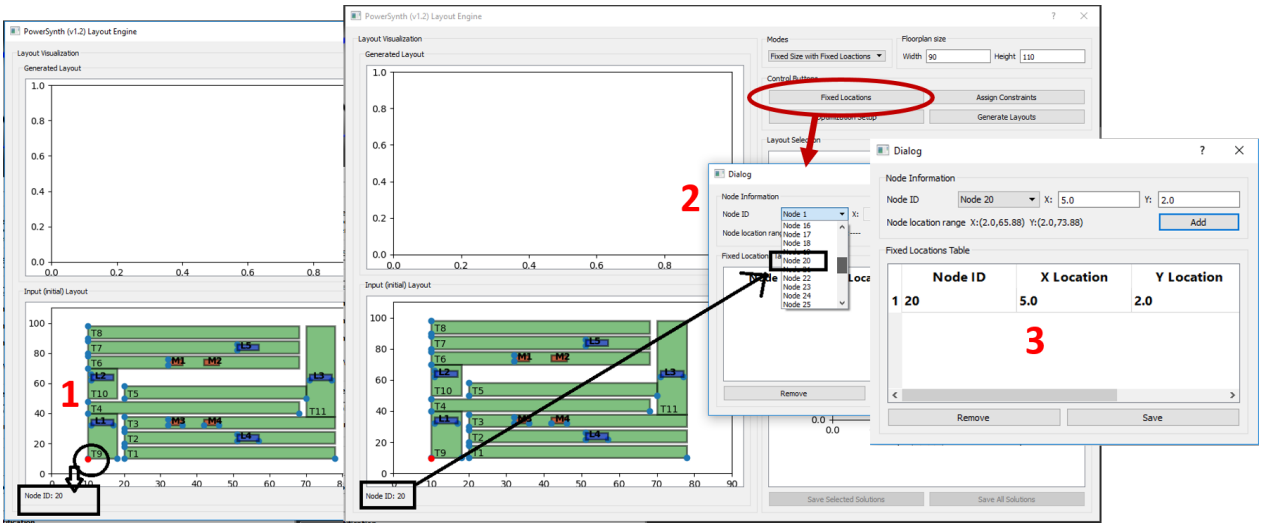


Figure 5: Fixed Location Setup

provide fixed location and then **"Save"** button needs to be clicked to apply the location values to the generated layout. A flow is shown in Fig. 5

At a time one mode of operation should be selected to generate solution. How to perform different mode operations is shown in the later part using an example.

2.6.4 Optimization Setup

To generate layout solution, the performance metrics should be chosen. To setup the measurement metrics **"Optimization Setup"** button needs to be clicked and a dialog (shown in Fig. 6) will appear.

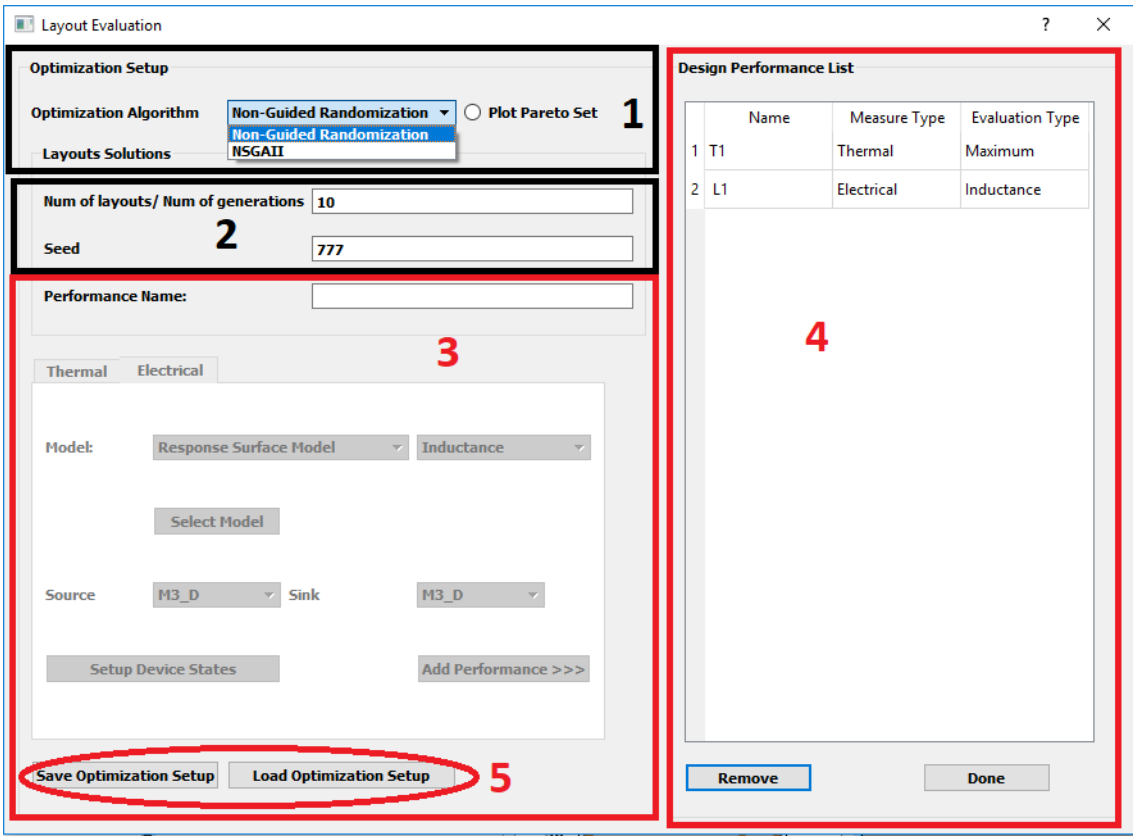


Figure 6: Optimization Setup Dialog

There are five parts of this dialog.

1. **Optimization Setup**: Here, there are two options for choosing an optimization algorithm: 1) Non-Guided Randomization, and 2) NSGAII. Currently, NSGAII is only available for **Fixed Size Layout** generation mode. The radio button should be checked for plotting the Pareto-front only otherwise the whole solution sets will be appeared on the solution browser.

2. **Layout Solutions**: For NSGAII, the number of generations and randomization seed should be provided, whereas for the non-guided randomization, the total number of layouts to be generated and seed need to be provided for all modes of layout generation except the **Minimum Size Layout** case.

3. **Performance Metrics Information**: In this version, two types of measurements are included: electrical (loop inductance, capacitance, and resistance) and thermal (maximum, average, and std. deviation). The measurement setup is similar to the steps described in 2.5.

4. **Design Performance List**: After describing each performance (i.e., electrical, thermal), upon clicking on the **"Add Performance"** button this list will be populated. In this version, at a time only two performance metrics can be chosen as the 3-D solution browser is not integrated in the beta-version.

5. **Save and Load Optimization Setup**: For a certain project, user can save optimization setup by clicking on **"Save Optimization Setup"** button just after declaring all design performances. Later, for the same project, user can load the saved optimization setup file using the button **"Load Optimization Setup"**.

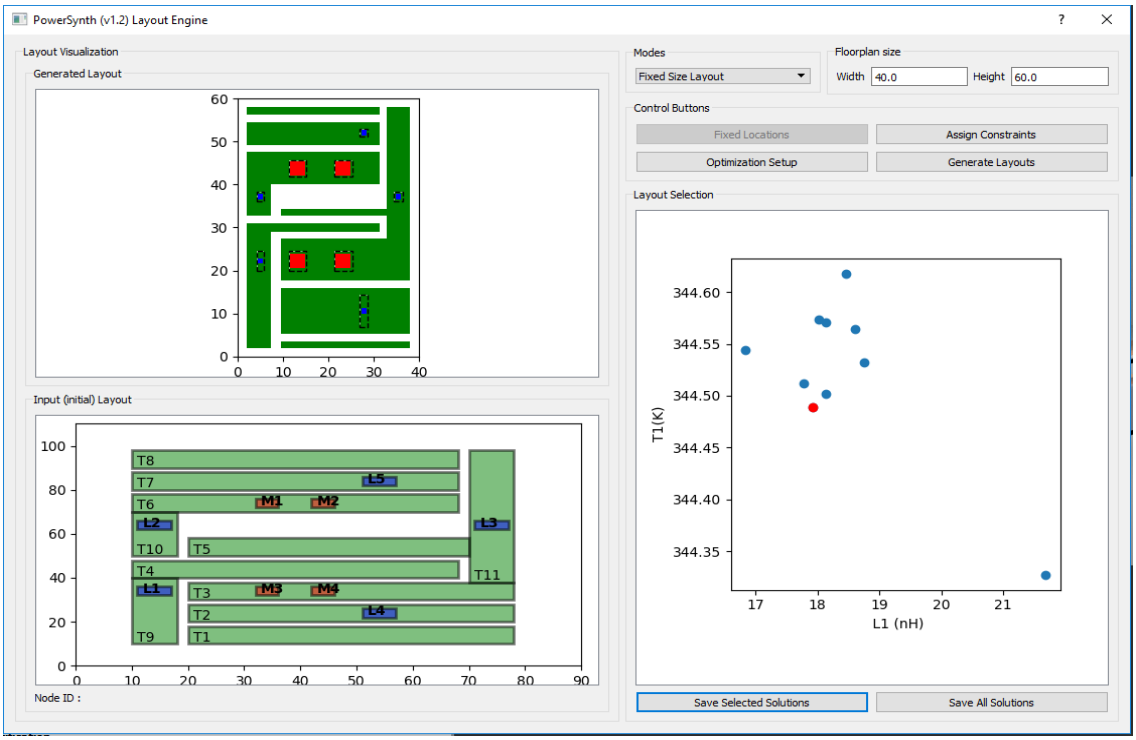


Figure 7: Generate Layouts

2.6.5 Generate Layouts

After completing each step stated above the **"Generate Layouts"** button should be clicked to generate solutions. Upon clicking this button, in the solution browser, solutions will appear as dots (shown in Fig. 7). In the figure, the red dot corresponding layout is appeared in the **Layout Visualization** section of the window. Upon clicking on each dot, the corresponding layout will appear there. As the widths and heights of all rectangles are varied, the devices and leads may not have a fixed dimension in the generated solutions. So, in the generated layout, the dotted lines show the possible room for that device or lead and the component is placed at the center of that room.

2.6.6 Saving Solutions

After layout generation, user can save each selected solution in the main window of PowerSynth by using **"Save Selected Solution"** button for further export operations (shown in Fig. 13). Also, all solution information can be saved as csv format upon clicking on the button **"Save All Solutions"**. In this case, for each solution, one csv file will be created having all performance values and layout information. Also, another csv file will save all performance data only.

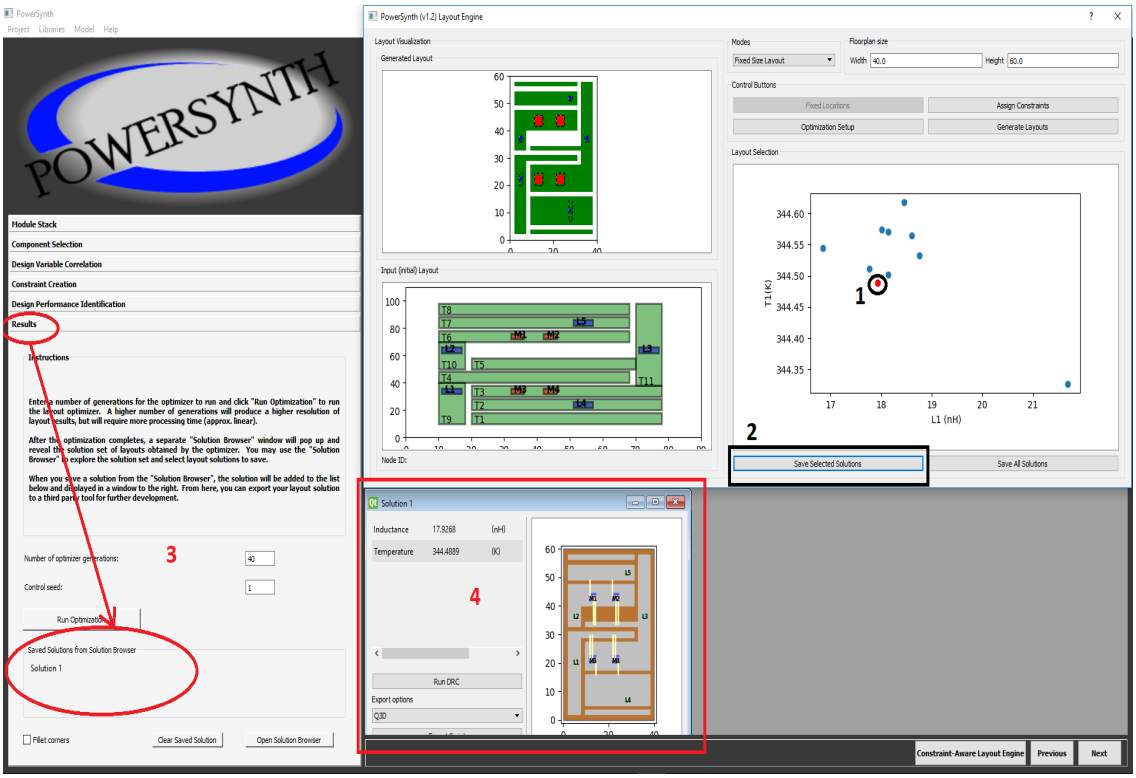


Figure 8: Saving Selected Solution

2.6.7 Test Case

The same case described in 2.2 is used for testing the new layout engine. The whole flow is shown sequentially.

To use new layout engine the input script must be in the same format described in 2.2. The name of the component must start with first letter of the component name (i.e., Trace–> T, MOSFET–> M, Lead–> L, and Diode–> D). In this version these four components are allowed to place in the floorplan.

Before clicking on the **"Constraint-Aware Layout Engine"** button user must complete each steps up to **"component selection"**. After that upon clicking on **"Constraint-Aware Layout Engine"** button the updated layout engine dialog will appear (shown in Fig. 9).

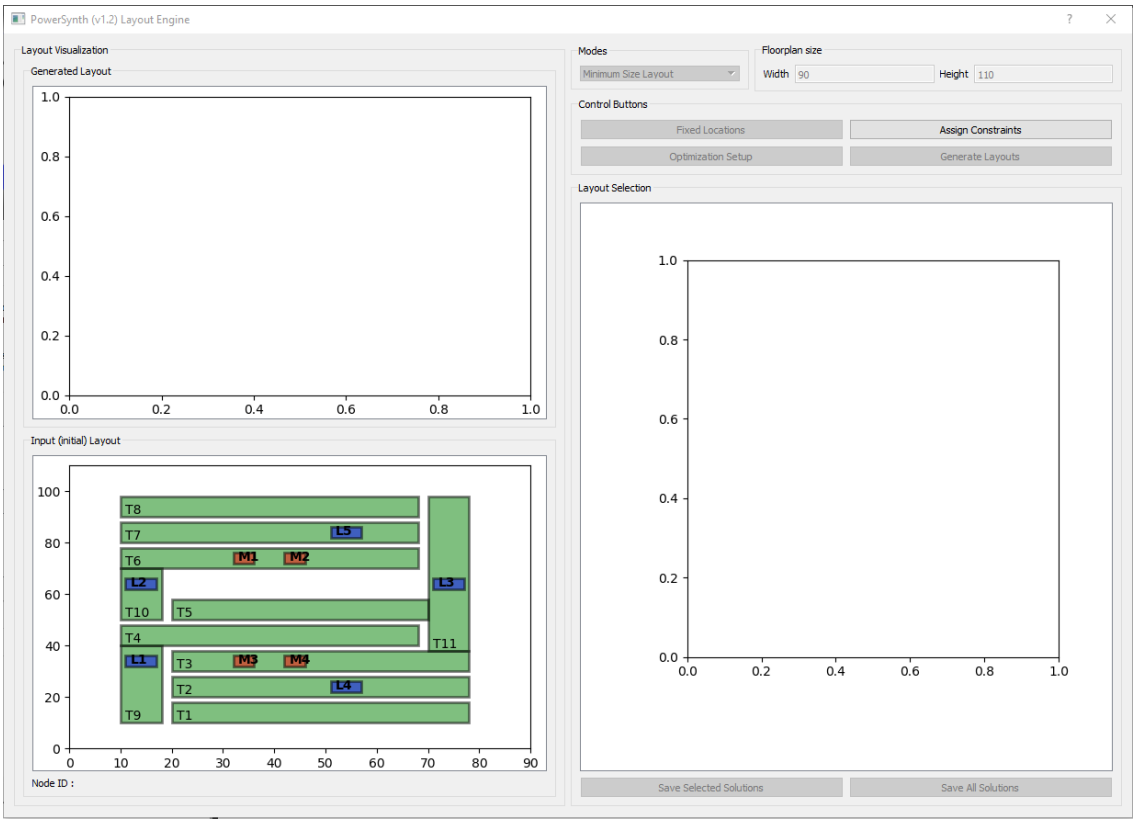


Figure 9: Constraint-Aware Layout Engine Dialog

At this stage, only **"Assign Constraints"** button is enabled. Click on the button and the constraint setup window (Fig. 10) will appear.

The constraint values are already inserted from technology library, so just click on the **"Apply to Layout"** button to implement the constraints.

Now, a mode of operation needs to be selected.

- **Mode 0: Minimum Size Layout**
  Choose **"Minimum size Layout"** from the drop down list on the "Modes" section. Then click on the **"Optimization Setup"** button to add performance metrics. Upon clicking the button, the optimization setup window will appear (shown in Fig. 11). Initially, only **"Performance Name"** field is editable as rest of them are fixed for Mode 0. Here, for thermal evaluation "T1" is entered as "Performance Name" and then click on the "Thermal" tab to choose thermal model and evaluation type. In this case, "FEM" model is chosen and "Maximum" temperature is picked for measurement. Then mark the check boxes corresponding the devices you want to measure. Here, all devices are considered. Then click on the **"Add Performance"** button to store the thermal measurement configuration in the "Design Performance List". Then again, the **"Performance Name"** field is populated with "L1" to measure inductance for the layout. Click on the "Electrical" tab to define electrical performance parameters. First, Model (Micro strip) and then type of measurement (Inductance) are chosen. In case of "Response Surface Model", **"Select Model"** button needs to be used to implement the model. Now, to calculate loop inductance one source and sink from the layout are chosen. In this example source is "L5" and sink is "L4". Then to complete the loop, **"Setup Device States"** button needs to be used to add proper connection (similar to shown in 2.5.5). Having all of the setup click on the **"Add**

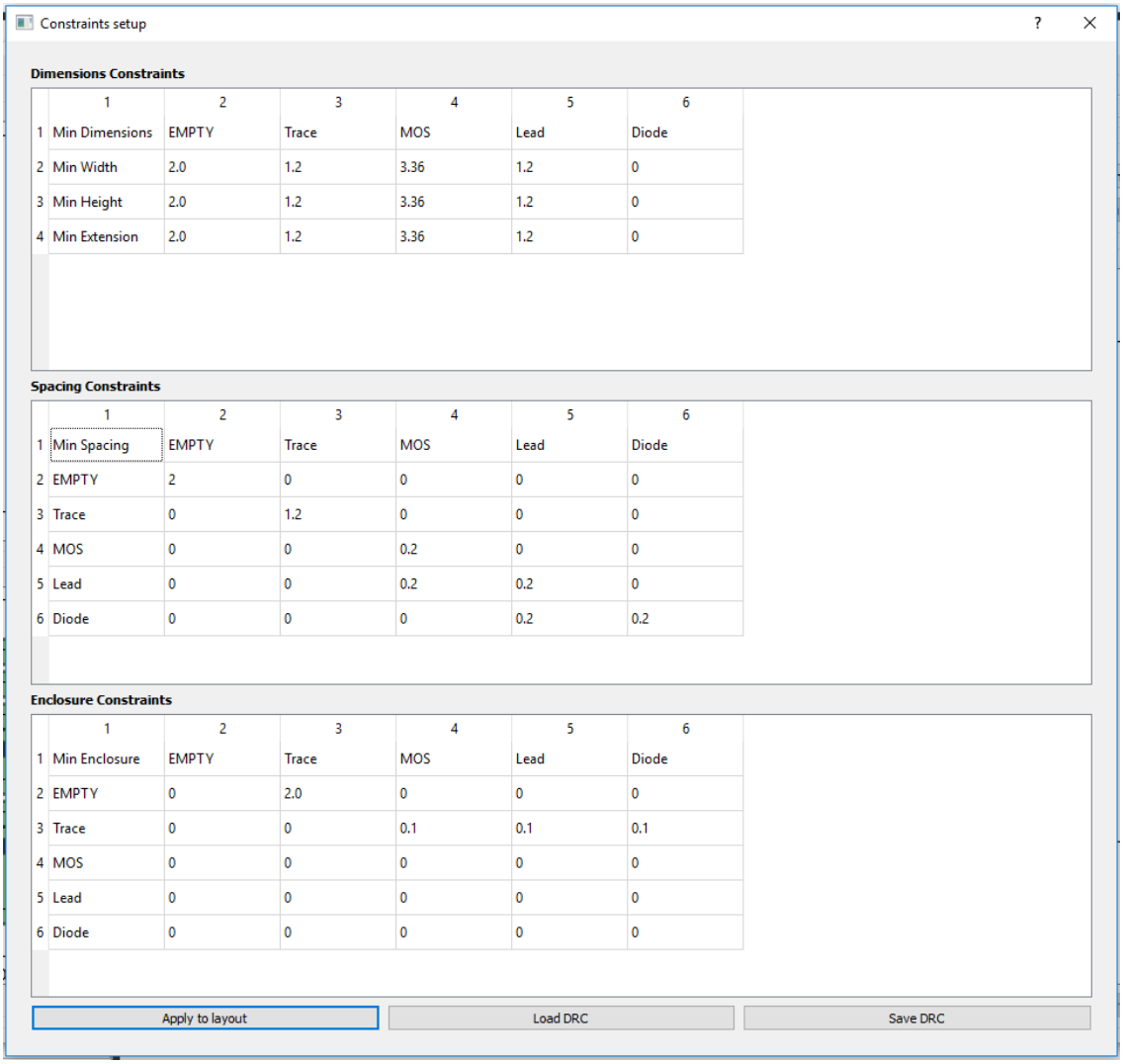


Figure 10: Constraint Table

Performance" button to add the measurement in the "Design Performance List". After each step the window status is shown in Fig. 11. Now click on the **"Done"** button. Also, the optimization setup can be saved by clicking on **"Save Optimization Setup"** button. Later the same setup file can be loaded for this case using the button **"Load Optimization Setup"**. After adding performances, click on the **"Generate Layouts"** button to have the minimum size layout. It will show one dot in the "Layout Selection" frame. Upon clicking on the dot, the **"Width"** and **"Height"** field will be populated with minimum size of the solution and "Generated Layout" frame will show the layout. The result is shown in Fig. 12.

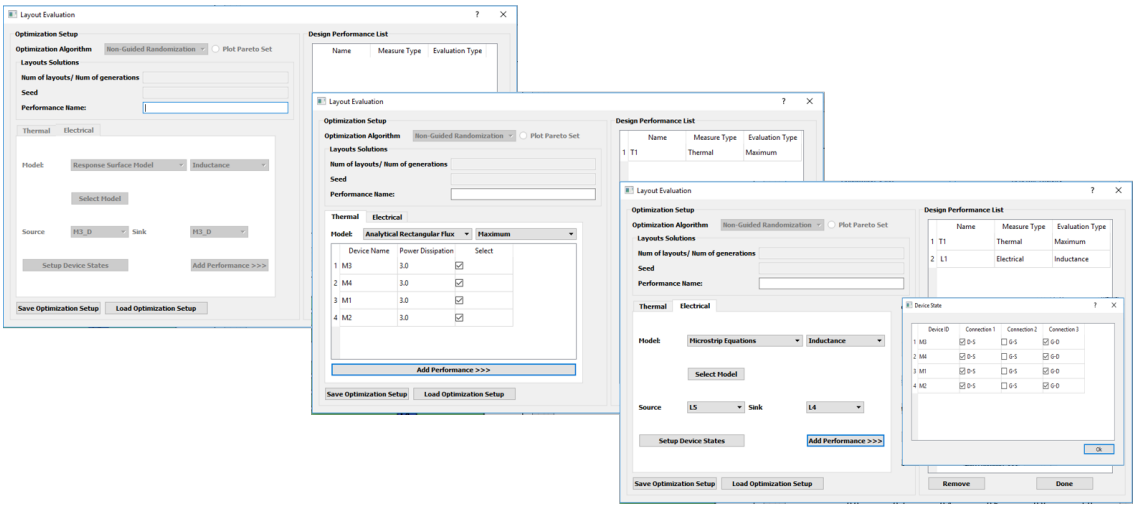


Figure 11: Mode 0 optimization setup flow

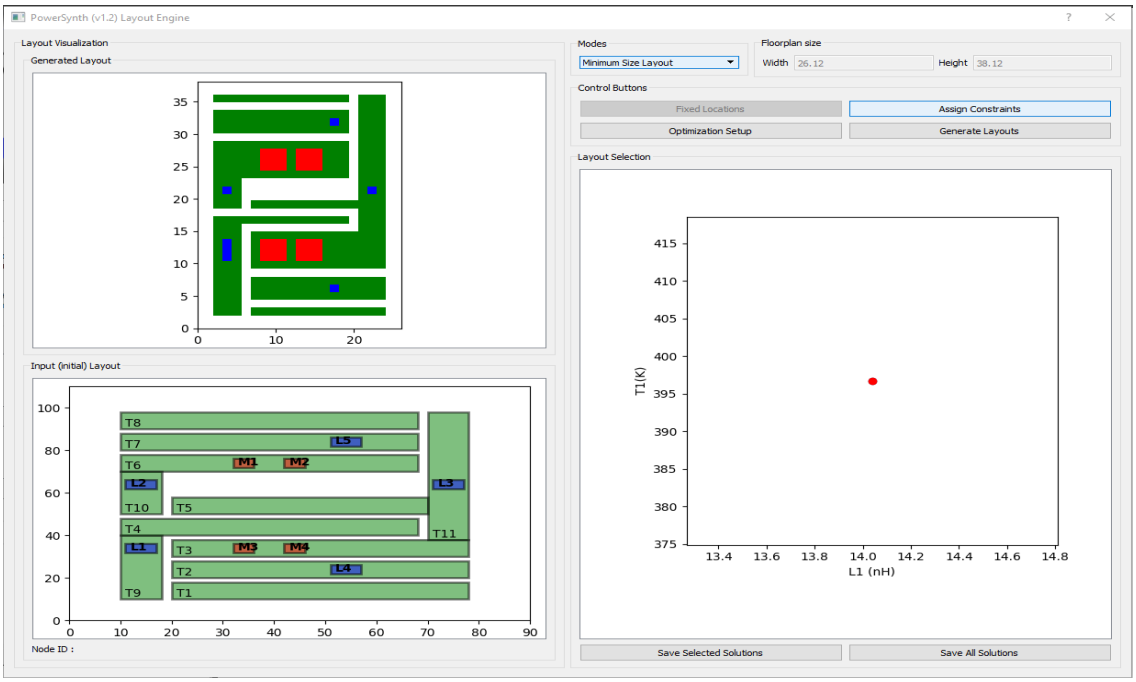
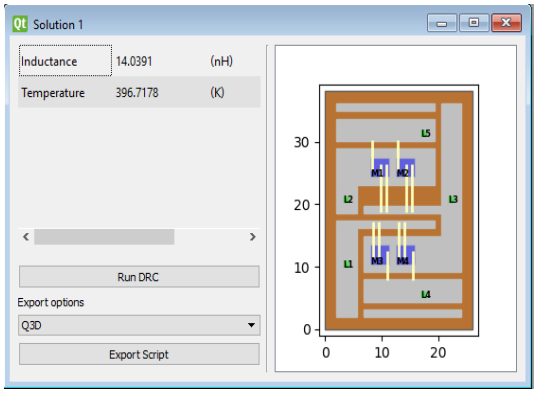


Figure 12: Minimum size Layout Solution

Using **"Save Selected Solutions"** button, the solution can be saved to main window (shown in Fig. 13 (a)) and using **"Save All Solutions"** button, the data can be saved to a csv file (shown in Fig. 13 (b)).

- **Mode 1: Variable Size Layout**
  To produce variable floorplan size layout solutions, select the **"Variable Size Layout"** from the drop down list on the "Modes" section. Now, click on the **"Optimization Setup"** button to input the "Number of layouts" to be generated and a randomization "Seed" in the corresponding fields. As performance metrics are already setup for this case, don't need to update that again. At this stage the "Optimization Setup" window will look like the same as



(a) Solution in main window

Size	Performance_1	Unit	Performance_2	Unit
[26.12000	14.03909006	(nH)	396.717826	(K)
Compone	x_coordinate	y_coordin	width	length
T8	2	34.92	17.32	1.2
T9	2	2	3.6	14.16
T6	2	23.16	17.32	5.76
T7	2	30.12	17.32	3.6
T4	2	16.16	17.32	1.2
T5	6.8	18.56	13.72	1.2
T2	6.8	4.4	17.32	3.6
T3	6.8	9.2	17.32	5.76
T1	6.8	2	17.32	1.2
M4	12.56	10.4	3.36	3.36
L4	16.92	5.6	1.2	1.2
L5	16.92	31.32	1.2	1.2
T10	2	18.56	3.6	4.6
L3	21.72	20.76	1.2	1.2
M3	8	10.4	3.36	3.36
L1	3.2	10.4	1.2	3.36
M1	8	24.36	3.36	3.36
M2	12.56	24.36	3.36	3.36
L2	3.2	20.76	1.2	1.2
T11	20.52	14.96	3.6	21.16

(b) Layout data in csv

Figure 13: Saving solution options

shown in Fig. 14.Also, to get the PAreto-front of the solutions, **"Plot Pareto Set"** can be checked.

Then click on the **"Generate Layouts"** button to generate 10 solutions. The result will appear on the solution browser and upon clicking on different dots, corresponding layout is shown in the "Generated Layout" frame. Also, "Width" and "Height" of the corresponding layout is shown in those fields. A sample layout is shown in Fig. 15. After getting all solutions, using the **"Save Selected Solutions"** button and **"Save All Solutions"** button those solutions can be exported.

- **Mode 2: Fixed Size Layout**
  Upon choosing this mode from the "Modes" options, the floorplan size needs to be provided in **"Width"** and **"Height"** fields. Then in the "Optimization Setup", there are two options: 1) Non-Guided Randomization and 2) NSGAII. For the first option, "Number of Layouts", and for the second option "Number of Generations" needs to be provided along with the "Seed". The setup looks like the Fig. 16. Now, after clicking the **"Done"** button,**"Generate Layouts"** button can be clicked. A new set of solutions will be appeared (shown in Fig. 17). Now, for NSGAII as "Optimization Algorithm" with 100 generations and seed=77, the solution set is shown in the Fig. 18(a) and corresponding Pareto-front is shown in the Fig. 18(b).

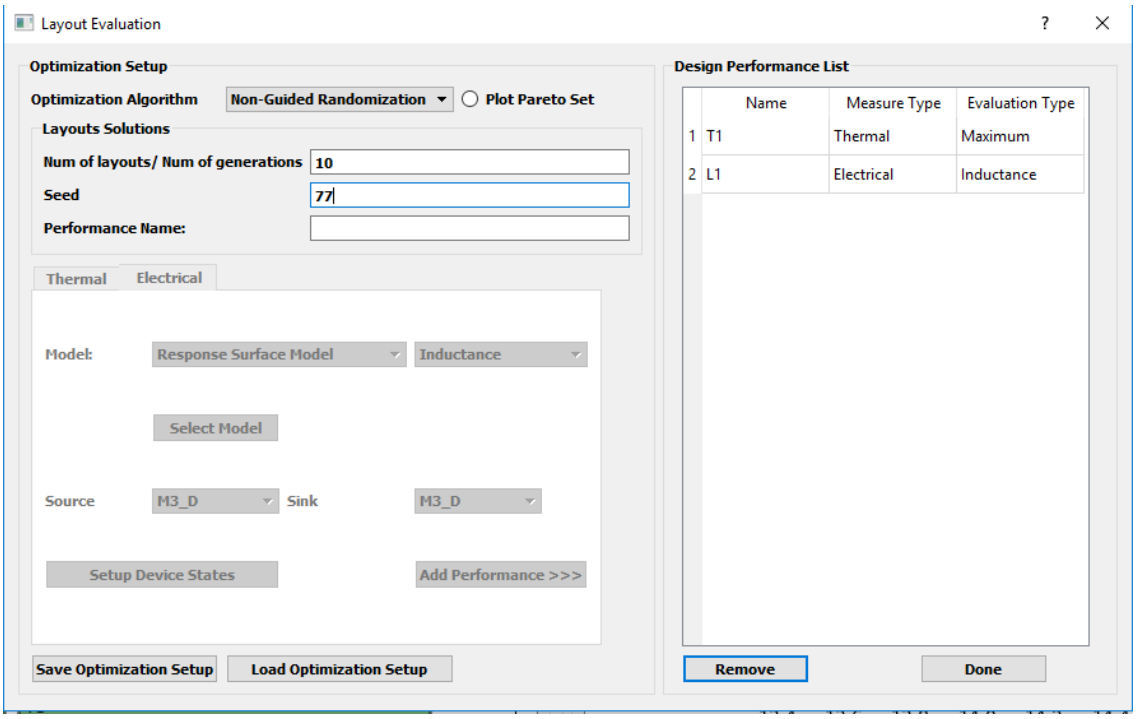


Figure 14: Mode 1 optimization setup

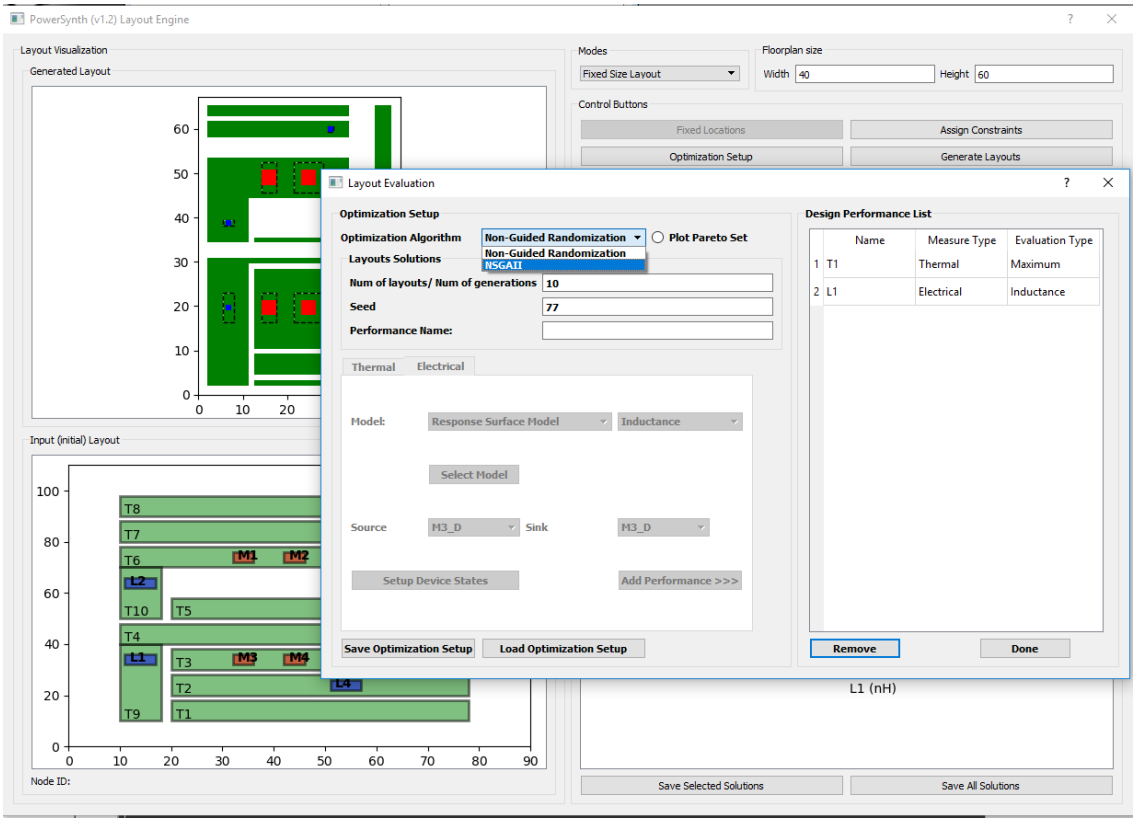


Figure 16: Mode 2 optimization setup

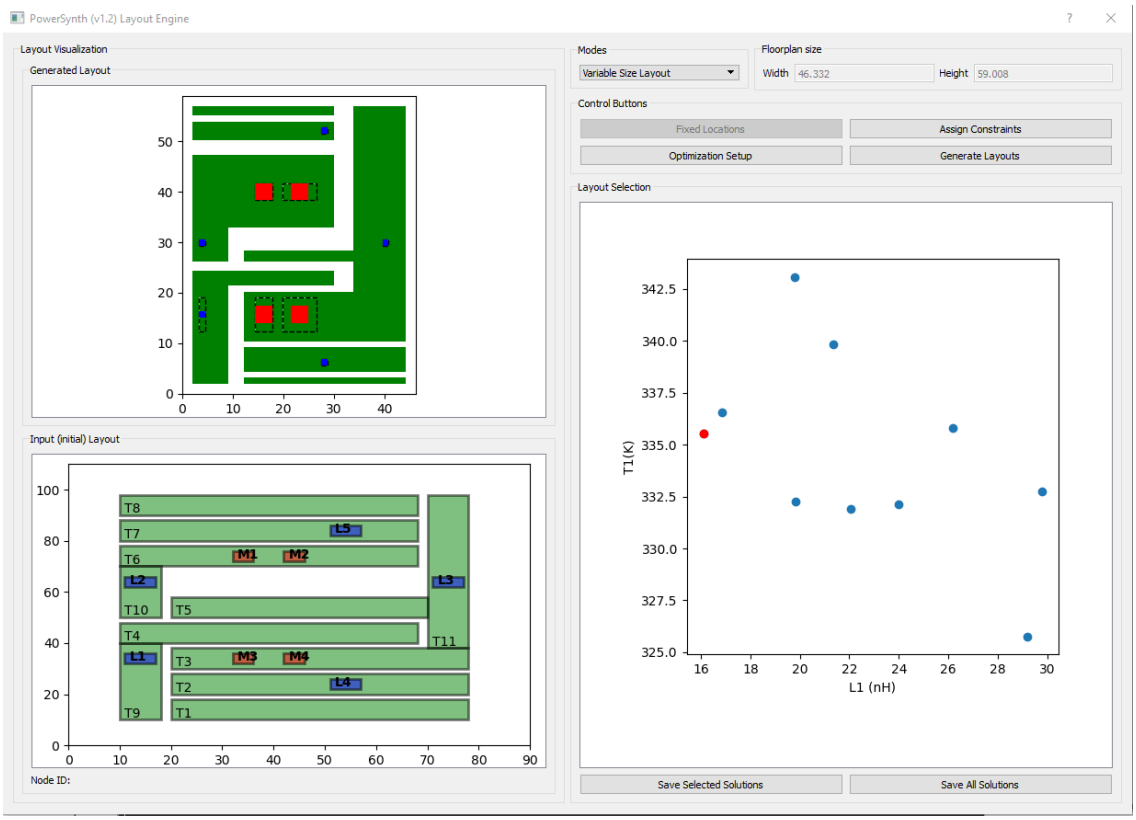


Figure 15: Mode 1 solution

- **Mode 3: Fixed Size with Fixed Locations**
  In this mode of operation, let's fix the leads "L4", and "L5" at certain x locations and leads "L2", and "L3" at certain y locations as shown in the table. as "L4", "L5" are correlated vertically and "L2","L3" are correlated horizontally, between two of them, we need to fix one.

Table 3: Desired fixed location of Leads

Components	x-location	Components	y-location
L4, L5	28	L3, L2	40

To setup locations according to the table, after selecting " **Fixed Size with Fixed Locations**" mode from the "Modes" click on the **"Fixed Locations"** button. Then map desired "Node ID" from the layout to the fixed location table. The table should look like the Fig. 19. Then click the **"Save"** button to store the given locations. The optimization setup is kept same as Mode 2 with non-guided randomization.

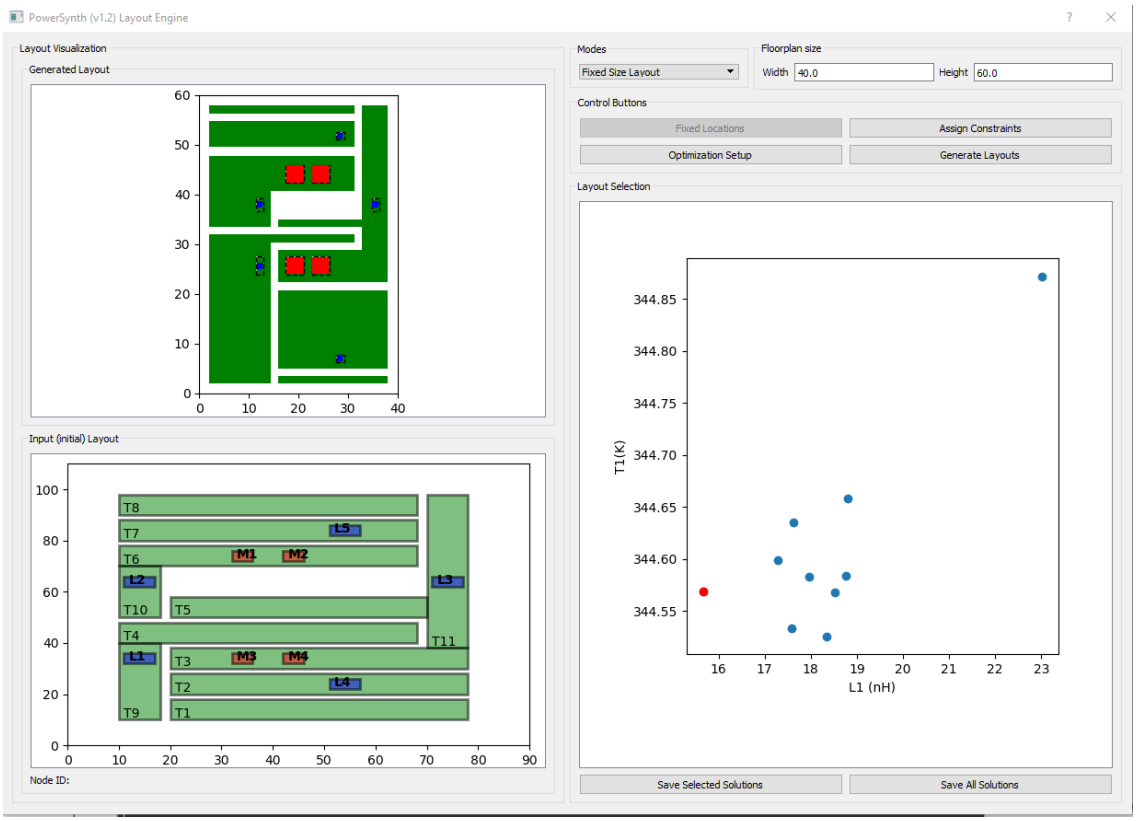


Figure 17: Non-Guided Randomization Solutions

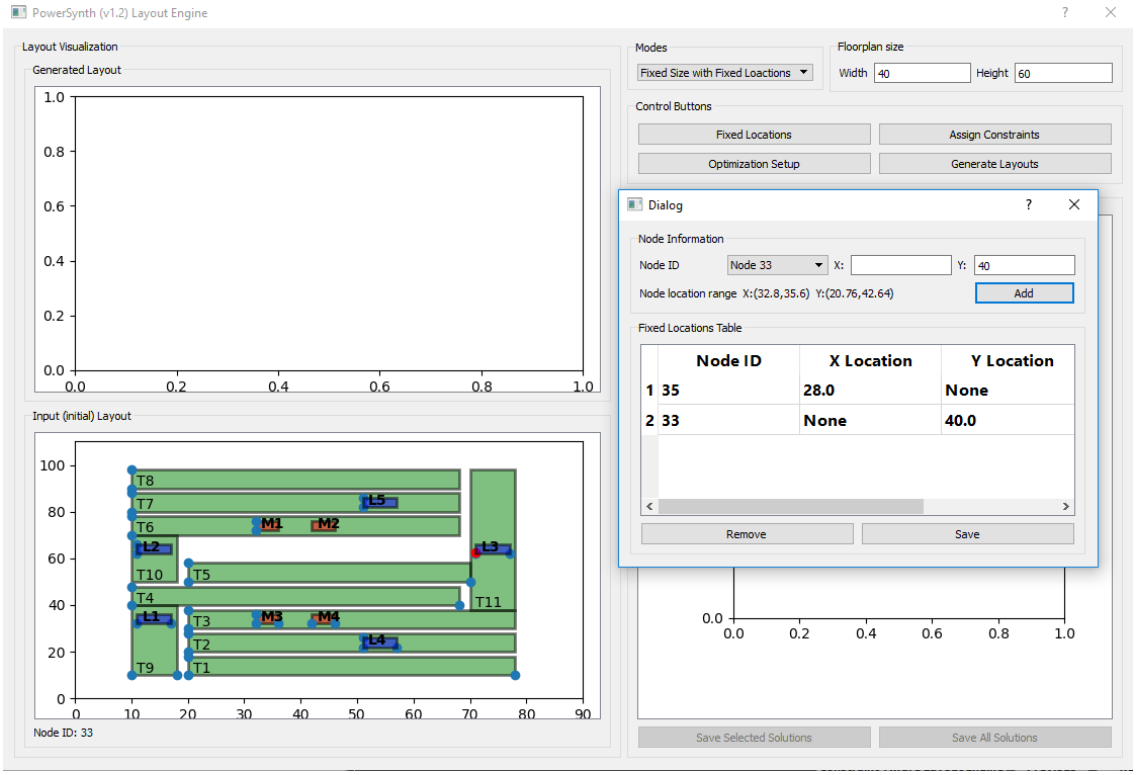
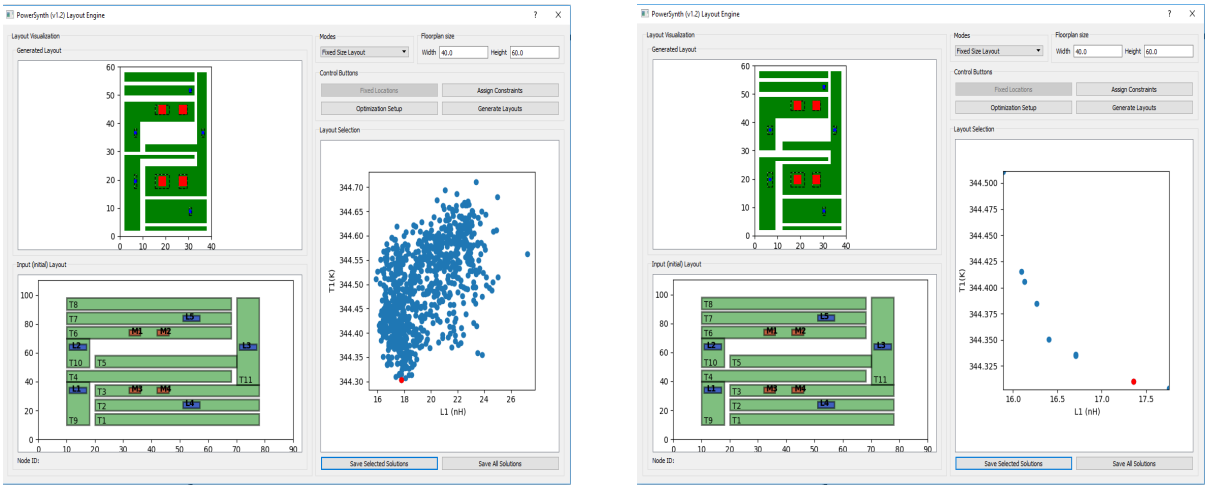


Figure 19: Fixed location setup



(a) Solution set for NSGAII (b) Pareto-front of (a)

Figure 18: Mode 2 optimization results for NSGAII

Upon clicking on **"Generate Layouts"** button the solution set will appear in the solution browser and look like the Fig. 20. Follow the dotted lines coordinate to find the match of the coordinates. In all of the solutions "L3" and "L2" have y-coordinate of 40, "L4" and "L5" have x-coordinate of 28.

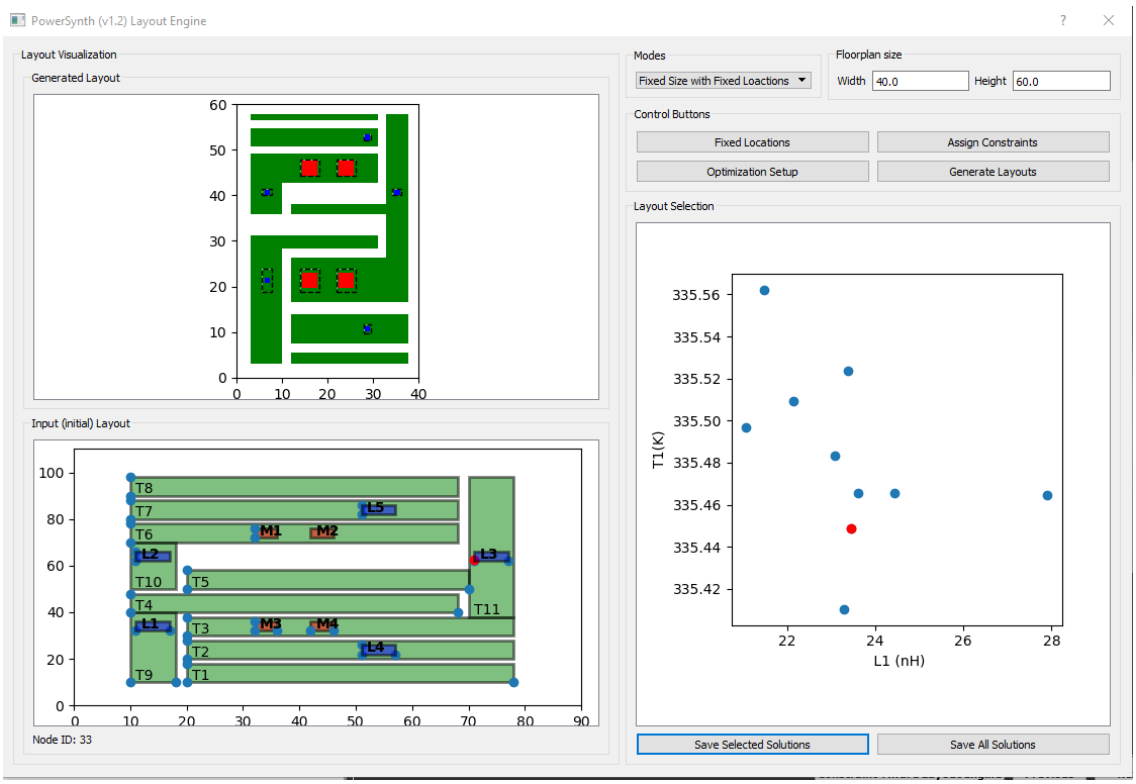
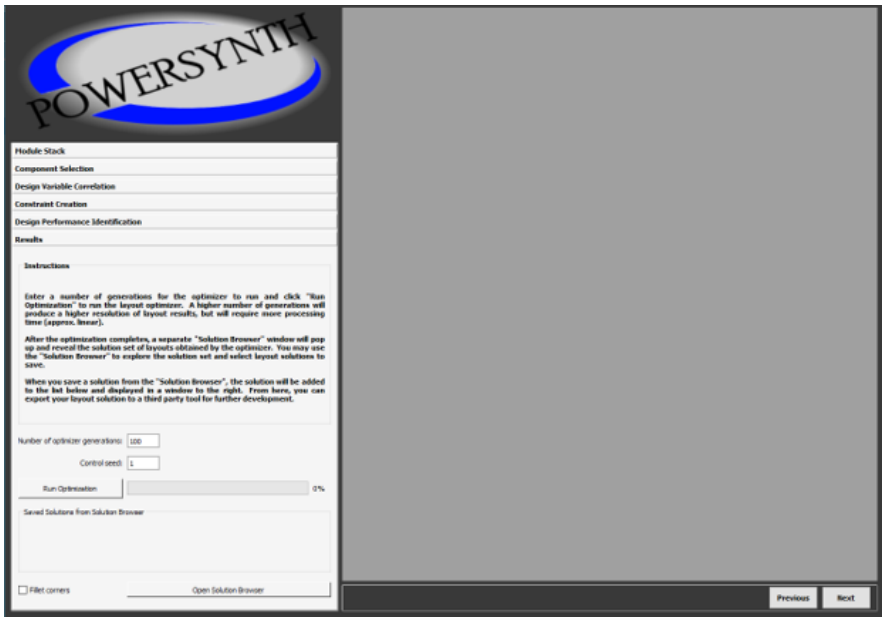


Figure 20: Mode 3 solutions

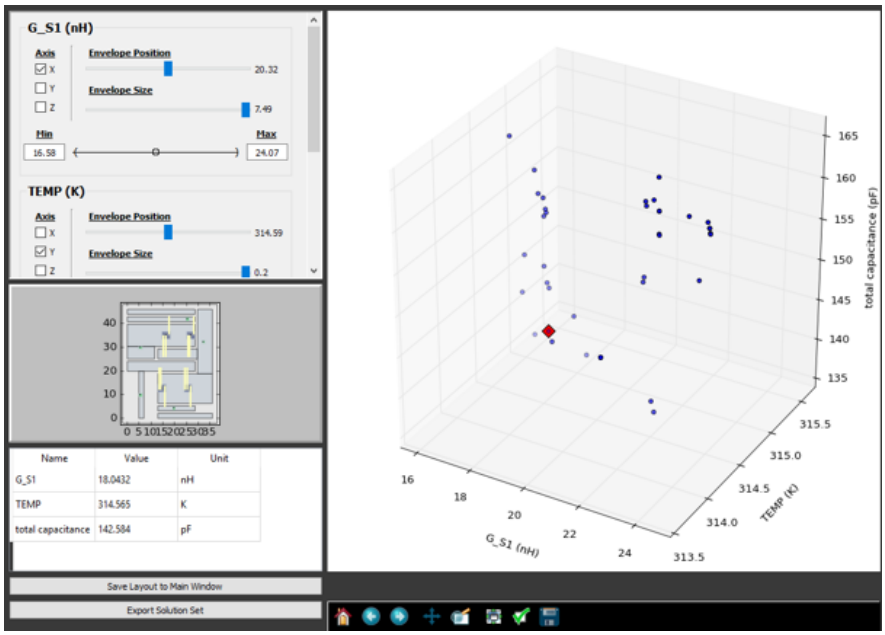
2.7 Optimization and Results

In this step, the user will need to specify number of generations. Since PowerSynth is based on a heuristic genetic algorithm, a higher number of generations will give you more solution choices. For this simple example 100 should be sufficient.

Click **Run Optimization.** The software will ask the user to save the project. Navigate to your specified project directory or any other directory to save this information. Note that the information from last run will be erased if it is not saved elsewhere. However, if the setup, number of generations, and the control seed (used for optimizer random generator) are kept the same, the final result will be the same. The runtime depends on number of generations specified. Upon completion, the solution window is opened as shown below.



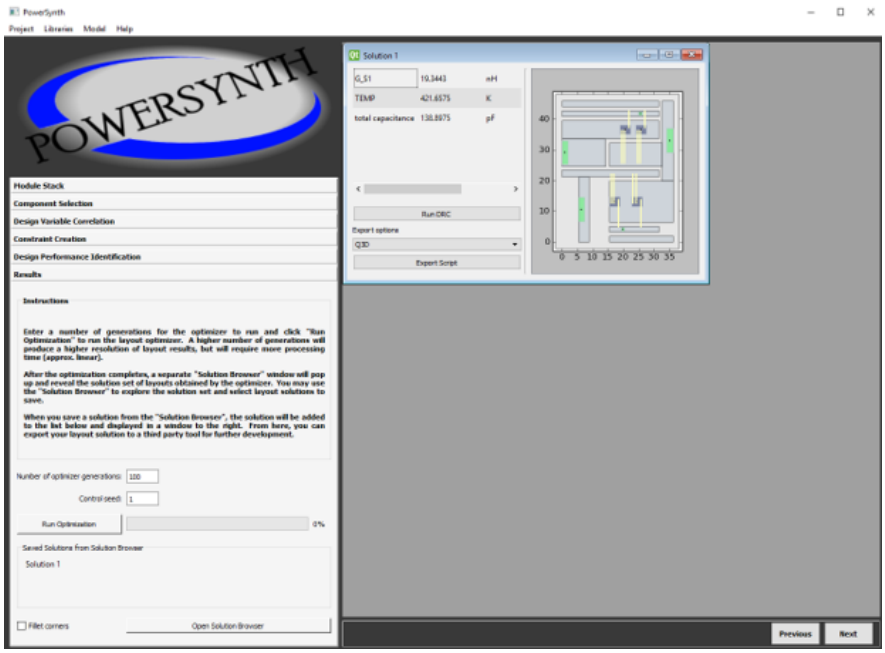
Check the console for progress updates. Once the optimizer routine is over, a new window should appear. Check your task bar if the window does not pop up. This is what the window may look like (depending on your unique solutions).



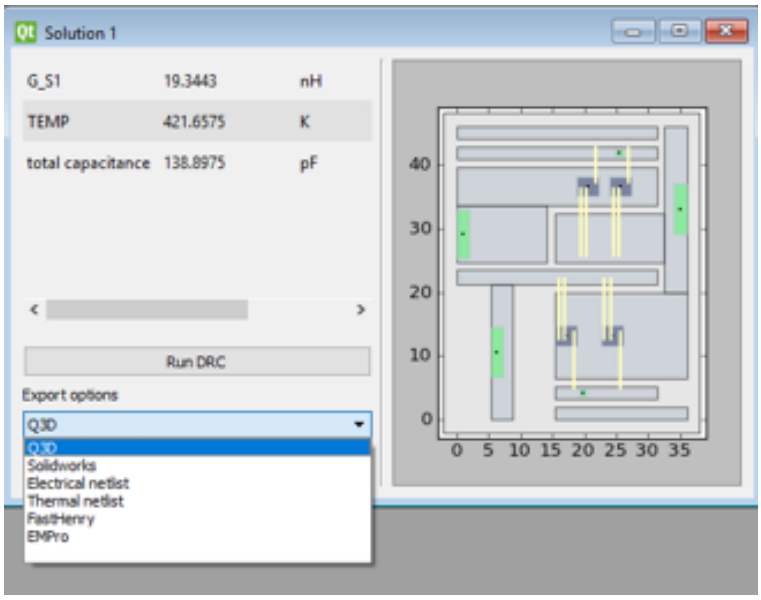
Left click and hold with your mouse inside the graph to rotate the graph. Choose the axes you wish to show using the pane on the left side (the scroll bar on the right will allow you to see more if they are not all visible). Each dot on the graph is a unique solution. Click on a solution and it will be colored red while displaying its preview on the left. Click on a different solution to see how the layout changes. The corresponding values for the objective functions are given below the layout preview.

To save the entire solution set, use the Export Solution Set button and save in .csv format. You can use Excel to open the file.

To save a solution for export into other tools or to see the E/T netlists, click on "Save Layout to Main Window." You can enter a name for this solution. In this case it is "Solution 1."



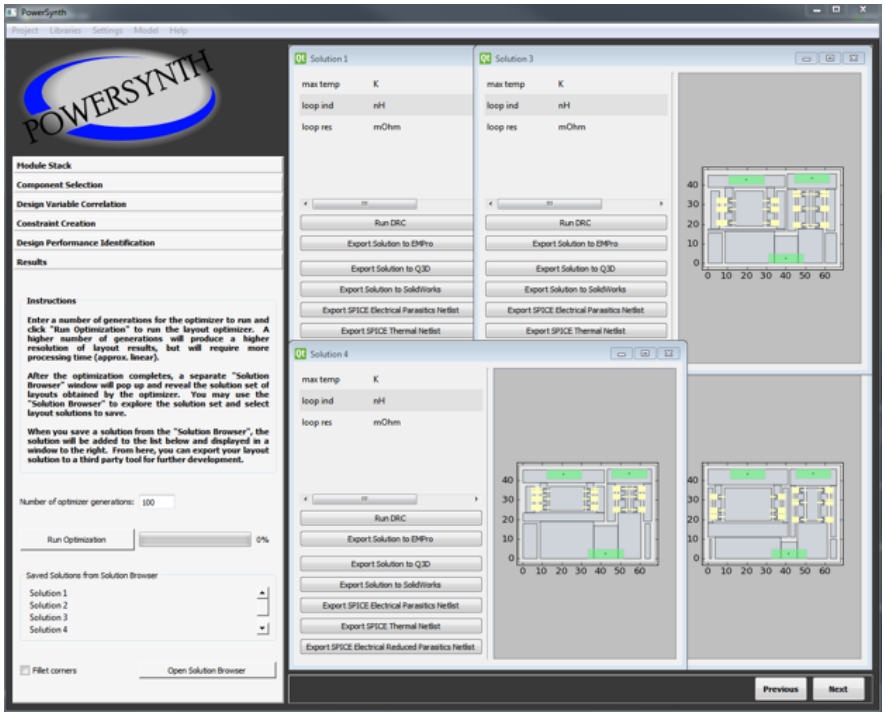
Under “Saved Solutions from Solution Browser” you can see Solution 1. The solution interface allows the user to export the solution to Q3D or SolidWorks. Other options allow the user to obtain the thermal and electrical netlists. The “Run DRC” button allow user to validate the design vs the Process Design Rule. To validate against different Process Design Rules see the section detailing the Process Design Rules Editor.



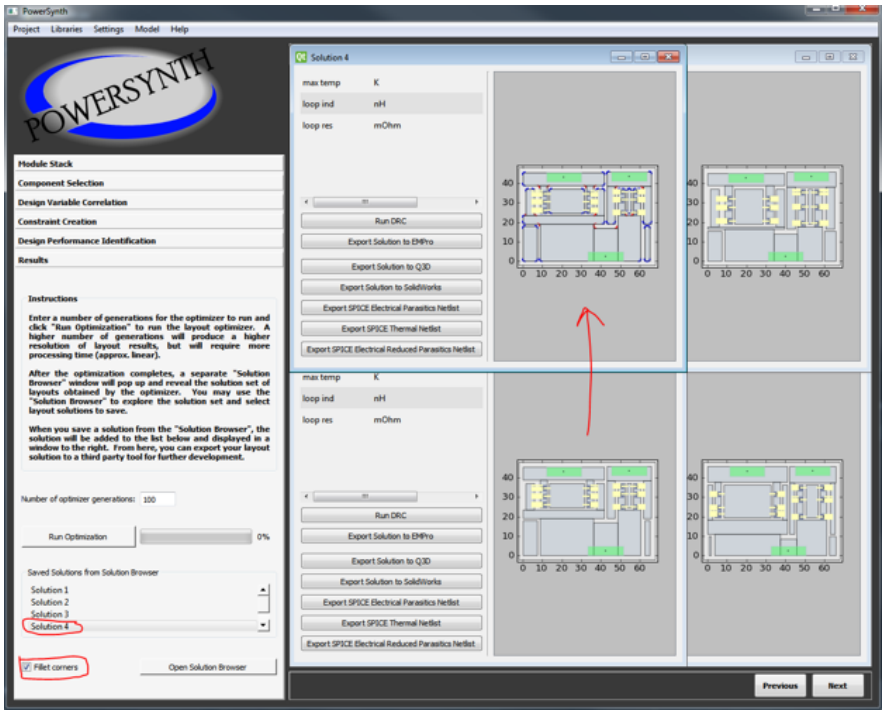
2.8 Post-Layout Optimization

PowerSynth allows the user to optionally apply corner correction to prevent electric field focusing and enhance thermo-mechanical reliability.

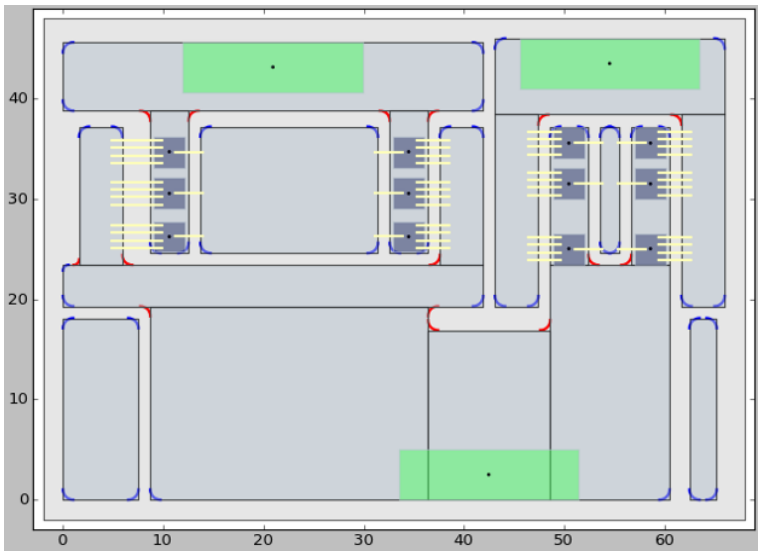
To apply fillets, in the Results tab, one may either check the 'Fillet corners' box, open the solution browser, and then select and save a solution, or alternatively, check the 'Fillet corners' box, and double click an existing solution under the 'Saved Solutions from Solution Browser' section to apply the fillets to an existing solution. For example, before applying fillets to solution 4:



And after applying fillets to solution 4, a new sub-window with the filleted solution appears:



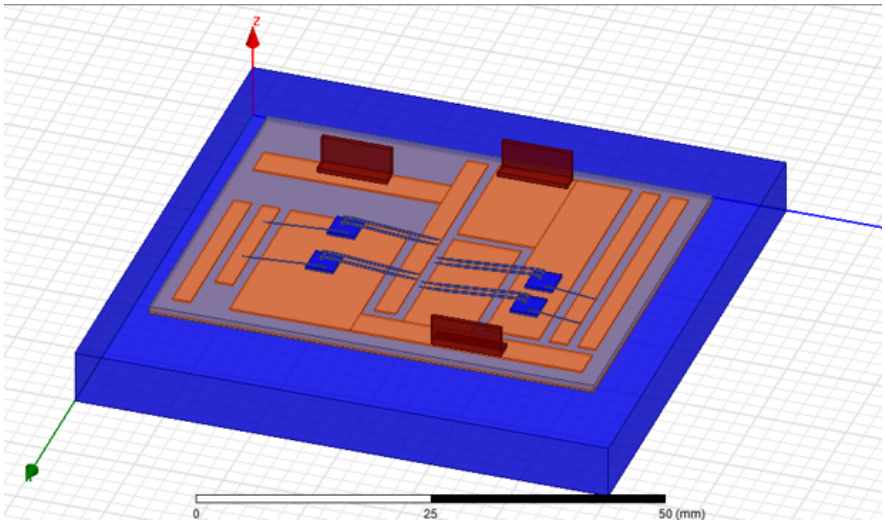
The arrow pointing up shows the same solution without fillets at the tail and with fillets at the head.



2.9 Exporting Saved Solutions

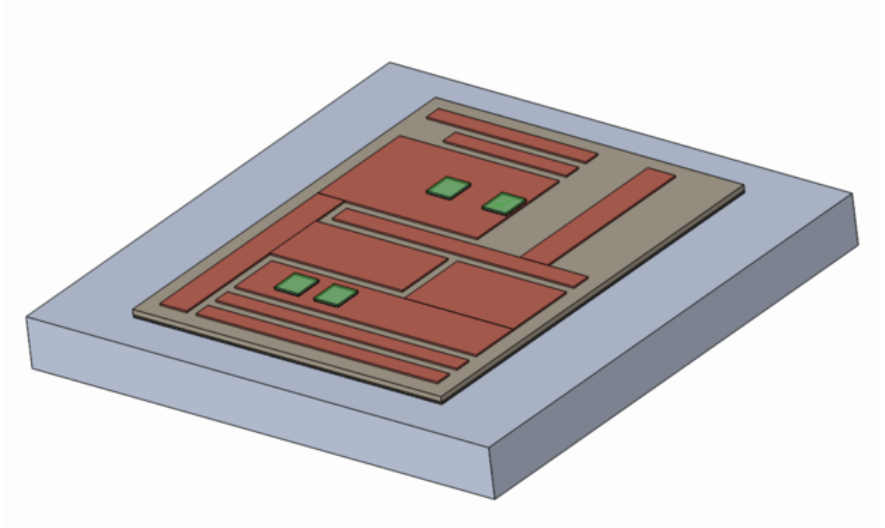
2.9.1 Export to ANSYS Q3D

Select `Q3D` from the export options list.Choose a directory, and create a file name. This will generate a .vbs file. In ANSYS Q3D open a new Q3D project. Go to **Tool-> Run Script** and load the exported .vbs file , the result is shown as below.



2.9.2 Export to SolidWorks

Choosing this option will open a dialog is asking the user for their SolidWorks version. After typing the version (e.g. 2016) for the SolidWorks. Hit ok and choose an output directory. A*.swp file is exported. However, this is not a binary file and can be somewhat troublesome to run thisscript. In SolidWorks, under Tools-> Macro select "New" (do not choose edit or run since the *.swp from PowerSynth is just a text file, not binary). Copy and paste the text from the exported SolidWorks file and paste into the macro code editor. Finally press "RUN" (green triangular button).



2.9.3 Export SPICE Netlist

This will export a text file for thermal and electrical netlists. An example electrical netlist is shown below. This sub circuit netlist is in a SPICE format which is compatible with many circuit simulators. The user can use this along with any device file to do the analysis.

```
.subckt X1 M3_D M4_D M1_D M2_D G_Low M2_S DC_neg \
        DC_plus 0Out M2_G G_High M3_S M3_G M4_S M4_G M1_S M1_G
R0 M3_D 0018 0.000128325620465
L0 0018 M4_D 6.90843508845e-10
C0 M4_D 0 8.36813864105e-12
R1 M3_D 0019 6.20960837181e-05
L1 0019 0020 2.68348440987e-10
C1 0020 0 4.26631730709e-12
R2 0021 0022 2.1073578894e-05
L2 0022 0023 4.59520341299e-11
R3 0021 0024 1e-06
L3 0024 0025 1e-12
R4 0021 0026 0.000395244569043
L4 0026 0027 1.09343343808e-09
C4 0027 0 1.08826978588e-12
R5 0025 0028 0.0291101534176
L5 0028 M3_G 4.33e-09
R6 0029 0030 0.000206772456704
...
```

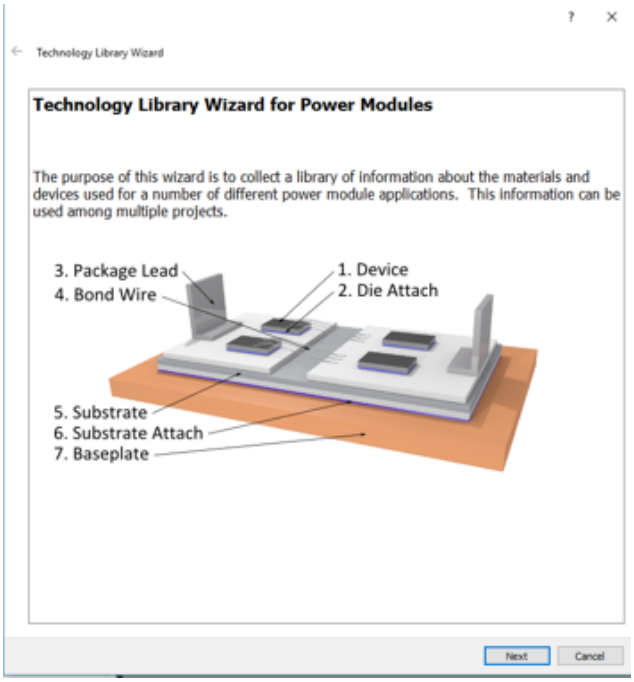
2.9.4 Export to Keysight EMPro

Please see the appendix for more detail.

3 Libraries and Editors

3.1 Technology Library Editor

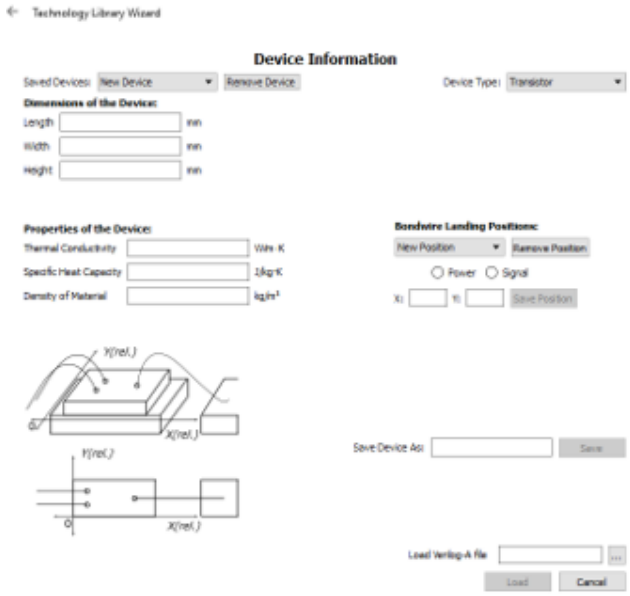
Under **Libraries**, select **Tech. Lib. Editor**. This allows the user to modify different technology files. The first page of the tech lib editor is shown on the right. Click the "Next" button to go through the pages of this wizard for creating or editing components.



3.1.1 Device Information

This page allows the user to input device information. This includes the dimension and material properties as well as the landing position for bondwires. The user can also input the Verilog A file for transistor/diode, while this information is not used in the optimization process, it will be included once the user exports the electrical netlist for a selected layout.

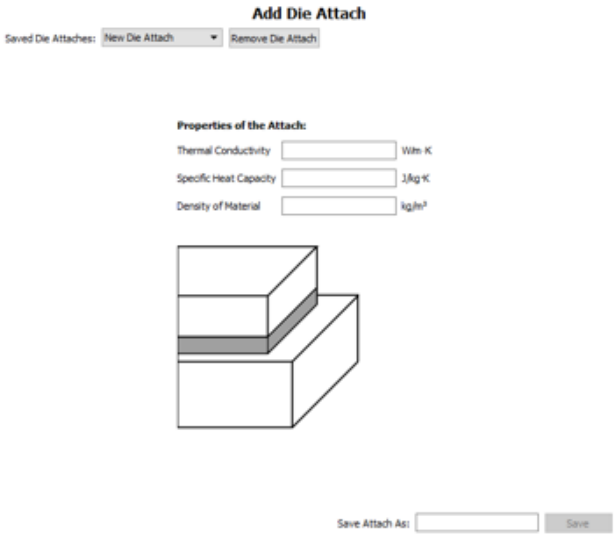
Click the "Save" button to save the new device to the library.



3.1.2 Add Die Attach

This page allows the user to assign the material properties for the die attach.

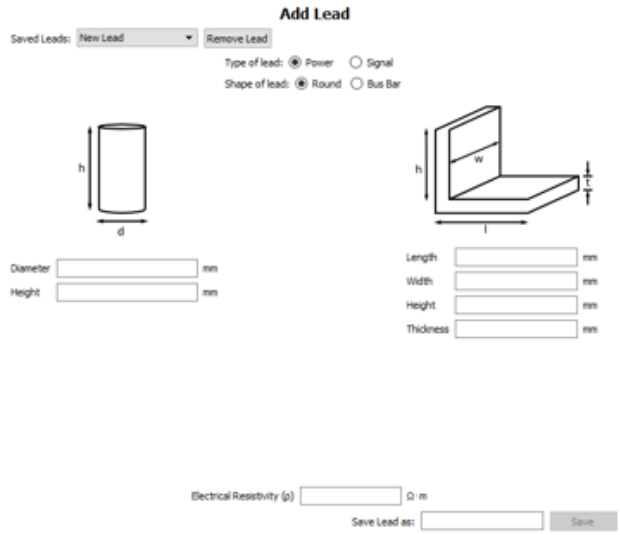
Click the "Save" button to save the new die attach file to the library.



3.1.3 Add Lead

This page allows the user to assign the material properties and dimensions for different lead types. In PowerSynth, this currently falls into two categories: “Round” and “Bus Bar leads.”

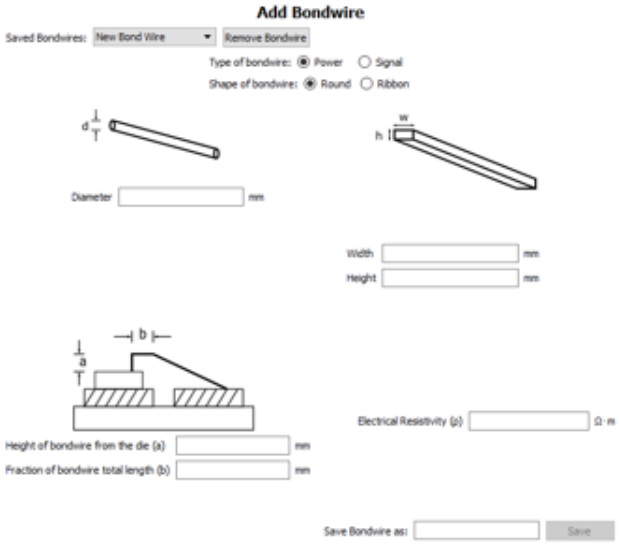
Click the “Save” button to save the new lead file to the library.



3.1.4 Add BondWire

This page allows the user to input dimensions and material properties for different bondwire types. The bondwire model is based on JDEC standards.

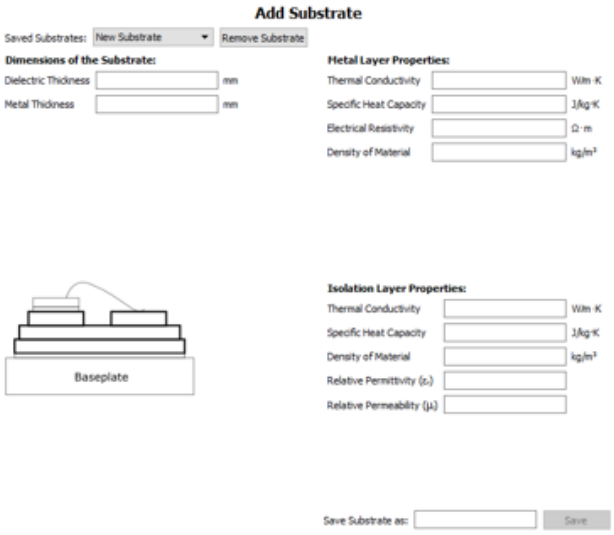
Click the “Save” button to save the new bondwire file to the library.



3.1.5 Add Substrate

This page allows the user to input dimensions and material properties for the metal and isolation layers of DBC or DBA substrates.

Click the "Save" button to save the new file to the library.



3.1.6 Add Substrate Attach

This allows the user to assign the material properties for the substrate attach.

Click the "Save" button to save the new substrate attach file to the library.

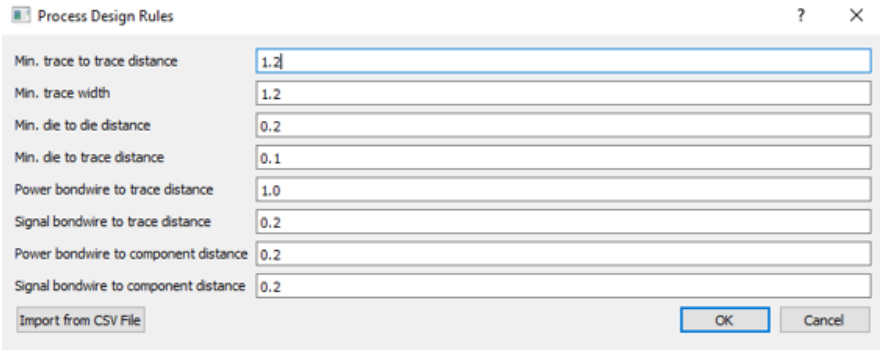
3.1.7 Add Baseplate

This allows the user to assign the material properties and dimensions for the baseplate.

Click the "Save" button to save the new baseplate file to the library.

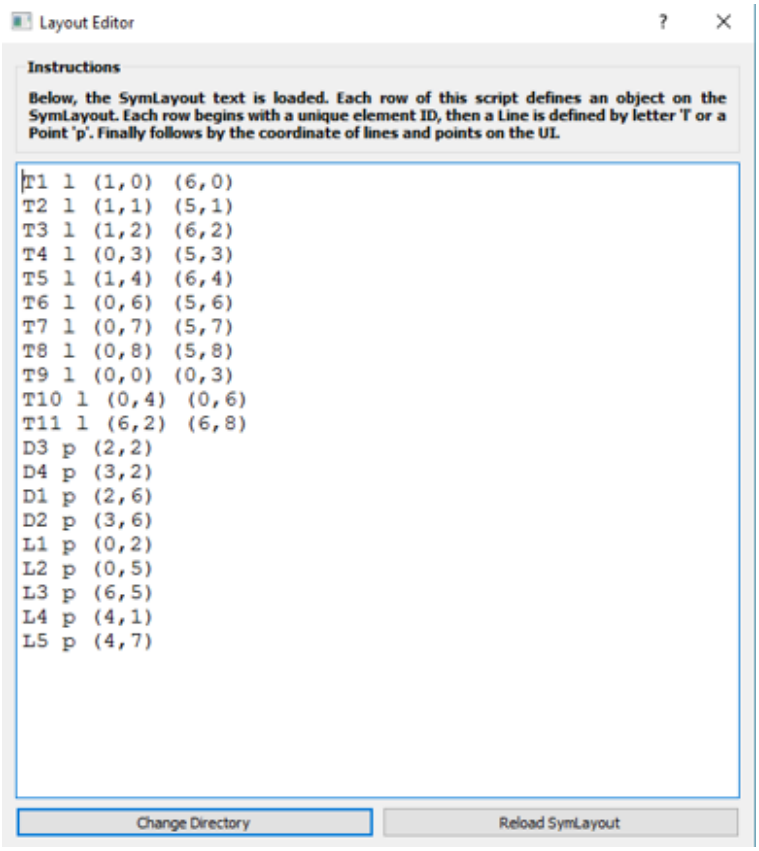
3.2 Process Design Rules Editor

The process design rules editor can be opened from **Project -> Design Rule Editors**. The interface below allows the user to input some design rule constraints for the layout. In the optimization process, candidate layouts that violate these rules are deemed as infeasible and will be eliminated from the solution space.



3.3 Layout Editor

The layout editor below allows the user to modify the symbolic layout. This can be found under **Project-> Open Layout Editor**. After the script is modified, click "Reload SymLayout" to load the new layout to the workspace.



4 PowerSynth-Related Publications

This section contains a list of all the current publications related to PowerSynth as of the date of this document.

1. P. Tucker, "SPICE netlist generation for electrical parasitic modeling of multi-chip power module designs," 2013.

2. B. W. Shook, A. Nizam, Z. Gong, A. M. Francis and H. A. Mantooth, "Multi-objective layout optimization for multi-chip power modules considering electrical parasitics and thermal performance," in *Control and Modeling for Power Electronics (COMPEL), 2013 IEEE 14th Workshop on*, 2013.

3. B. W. Shook, Z. Gong, Y. Feng, A. M. Francis and H. A. Mantooth, "Multi-chip power module fast thermal modeling for layout optimization," *Computer-Aided Design and Applications*, vol. 9, pp. 837-846, 2012.

4. B. W. Shook, "The Design and Implementation of a Multi-Chip Power Module Layout Synthesis Tool," 2014.

5. J. Main, "A Manufacturer Design Kit for Multi-Chip Power Module Layout Synthesis," 2017.

6. Q. Le, T. Evans, S. Mukherjee, Y. Peng, T. Vrotsos and H. A. Mantooth, "Response surface modeling for parasitic extraction for multi-objective optimization of multi-chip power modules (MCPMs)," *in Wide Bandgap Power Devices and Applications (WiPDA), 2017 IEEE 5th Workshop on*, 2017.

7. Q. Le, S. Mukherjee, T. Vrotsos and H. A. Mantooth, "Fast transient thermal and power dissipation modeling for multi-chip power modules: A preliminary assessment of different electro-thermal evaluation methods," in *Control and Modeling for Power Electronics (COMPEL), 2016 IEEE 17th Workshop on*, 2016.

8. Z. Gong, "Thermal and electrical parasitic modeling for multi-chip power module layout synthesis," 2012.

9. T. M. Evans, Q. Le, S. Mukherjee, I. Al-Razi, T. Vrotsos, Y. Peng and H. A. Mantooth, "PowerSynth: A Module Layout Generation Tool," in *IEEE Transactions on Power Electronics*, vol 34. no 6, pp. 5063-5078, June 2019, doi: 10.1109/TPEL.2018.2870346. *Highlighted Article.*

10. S. Mukherjee et al, "Toward Partial Discharge Reduction by Corner Correction in Power Module Layouts," in *Control and Modeling for Power Electronics (COMPEL)*, 2018.

11. I. Al Razi et al, "Constraint-Aware Algorithms for Heterogeneous Power Module Layout Synthesis and Reliability Optimization," in *in Wide Bandgap Power Devices and Applications (WiPDA), 2018 IEEE 6th Workshop on*, 2018.

12. T. M. Evans, S. Mukherjee, Y. Peng and H. A. Mantooth, "Electronic Design Automation (EDA) Tools and Considerations for Electro-Thermo-Mechanical Co-Design of High Voltage Power Modules," 2020 *IEEE Energy Conversion Congress and Exposition (ECCE)*, Detroit, MI, USA, 2020, pp. 5046-5052, doi: 10.1109/ECCE44975.2020.9235818.

5 Appendix 1: EMPro Export

5.1 Introduction

The EMPRo Export feature in PowerSynth allows a user to export designs generated by PowerSynth for further analysis in Keysight's EMPro and ADS. This document presents an overview of the capabilities available after export as well as the process in doing so. Furhtermore, while this document provides a walkthrough of some of the key points for usage, it is assumed that the user is familiar with Keysight EMPro and ADS.

5.1.1 Functionality

This feature of PowerSynth takes a layout design from PowerSynth and generates a script to be run in EMPro. This script automatically defines the simulation type and frequency range, draws all of the design components and assigns materials, and sets up all of the requisite ports for simulation.

Exporting a design generated by PowerSynth to EMPro grants the user several cababilities for futher analysis of a given design. These include, but are not limited to:

- S-Parameter modeling of a design
- Near and far field visualization
- Export to ADS for:
  - Incorporation of Verilog-A device models
  - Transient switching analysis
  - Conducted EMI analysis
  - Near field visualization of transient results

5.1.2 Caveats and Limitations in Current Release

Currently PowerSynth assumes a module buildup consisting of:

- Heat spreader
- Substrate attatch
- DBC-type substrate (Cu-ceramic-Cu)
- Die attatch
- Device
- Wire bonds

This stackup is fixed in the current release and so the resulting model in EMPro reflects this. However, future versions of the tool will allow greater flexibility.

During the export, the thin attatchment layers are neglected to allow for easier meshing. Additionally, devices are not physically modeled. Instead, pads corresponding to those of the devices are inserted and wire bonds are connected to them. This is done so that only the parasitics of the module are extracted. Later, in circuit simulators such as ADS, the user can insert a device model of their choosing into the corresponding ports inside the extracted layout parasitics model.

Table 4: Component Material Assignments

Component	Material
Heat Spreader	Aluminum
Traces	Copper
Ceramic	Alumina
Device Pads	PEC
Wire Bonds	Aluminum

Furthermore, materials are automatically assigned to components as shown below, but may be easily changed within EMPro.

As noted above, many of these issues related to the current limitations and assumptions in PowerSynth are currently being revised. As of this writing, a new layout engine is being developed to overcome them and provide even greater flexibility in design. The reader's patience is greatly appreciated as development of PowerSynth continues.

Finally, this feature and document were made using the 2017 versions of EMPro and ADS. Future version compatibility will be ensured. However, compatibility with previous versions has not been tested nor can it be guaranteed.

5.2 Exporting from PowerSynth

Once a design has been selected from PowerSynth, exporting the EMPro script is a straight forward task. This section will cover the steps necessary for exporting only and assumes that PowerSynth has been setup, run, and the user has been presented with design solutions. Please refer to the main PowerSynth manual for information on how to perform these steps.

5.2.1 Selecting a Design

After PowerSynth completes the optimization routine, the user is presented with a solution as shown in Figure 21. From here, select a design and click on the button "Save Layout to Main Window" then return to the main PowerSynth window.

5.2.2 Saving the EMPro Script

In the main window, find the sub-window containing the design that is to be exported. From there, click on the button titled "Export Solution to EMPro" as shown in Figure 22. You will then be presented with a dialog box asking you to specify a name and location for the python script which can be used in EMPro like in Figure 23. The next section will cover how to import the file into EMPro.

5.3 Importing to EMPro

Upon saving EMPro script, the file can be opened with EMPro to setup the model and simulation. The following steps provide more detail on these tasks.

5.3.1 Creating a New Project

First, open EMPro. Then select "New Project" from the file menu. Here, choose "Generic FEM, design in mm" as shown in Figure 24. Set the frequency ranges to be simulated and press "OK."

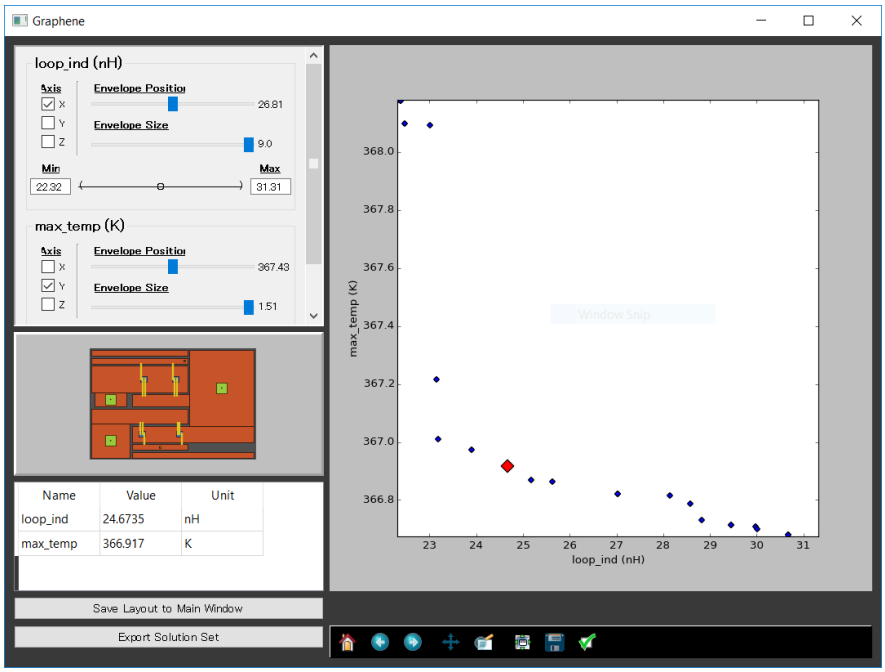


Figure 21: Selecting a design from the solution browser window and saving to the main PowerSynth one.

5.3.2 Running the Script

Next, open the scripting pane in EMPro. From here choose to create a new script from the file menu as in Figure 25. Open the script in an external text editor and copy the entire code, then paste it into the new script window in EMPro that was just created. Finally, press the execute script button (Figure 26) and wait until it completes.

Once the script has finished running, you will see the components fully modeled an populated in the main window like in Figure 27. Now the project is ready for simulation.

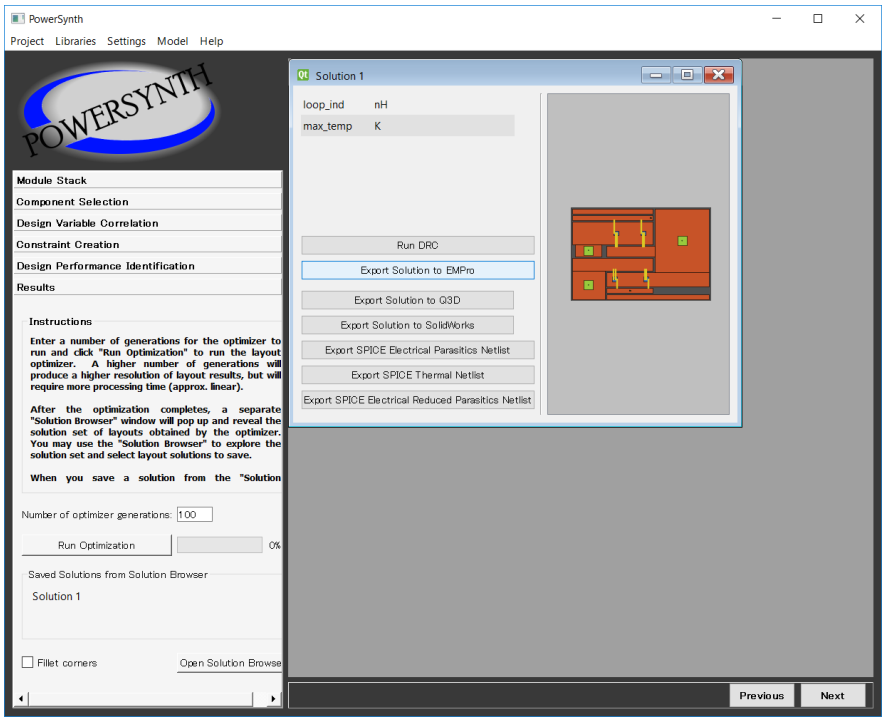


Figure 22: Exporting the design to an EMPro Python script.

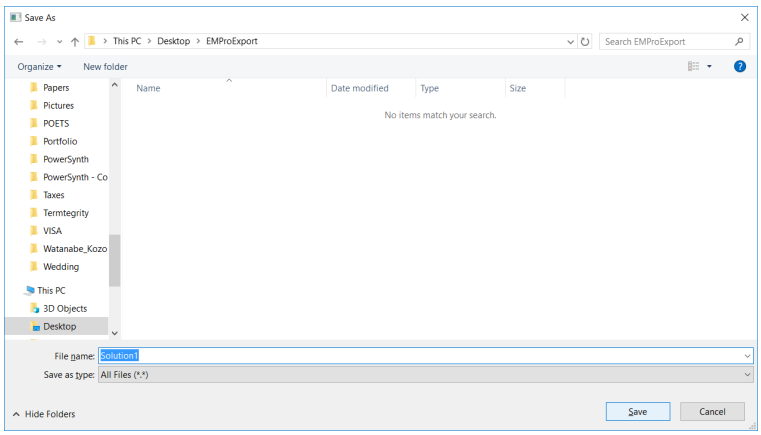


Figure 23: Specifing the file name for export to EMPro.

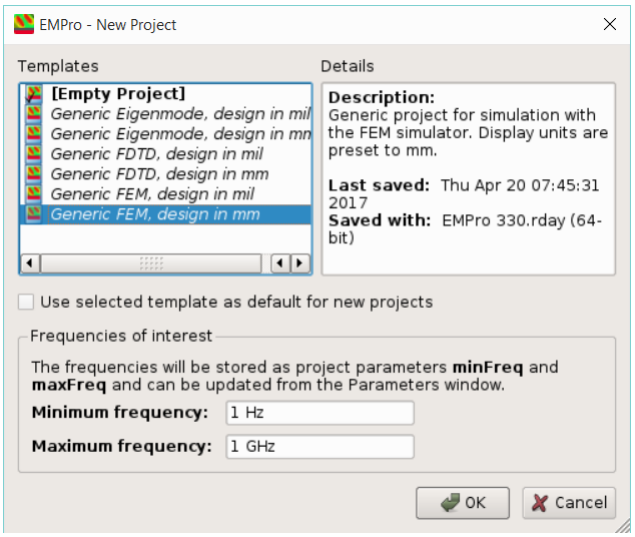


Figure 24: Creating a new EMPro Project.

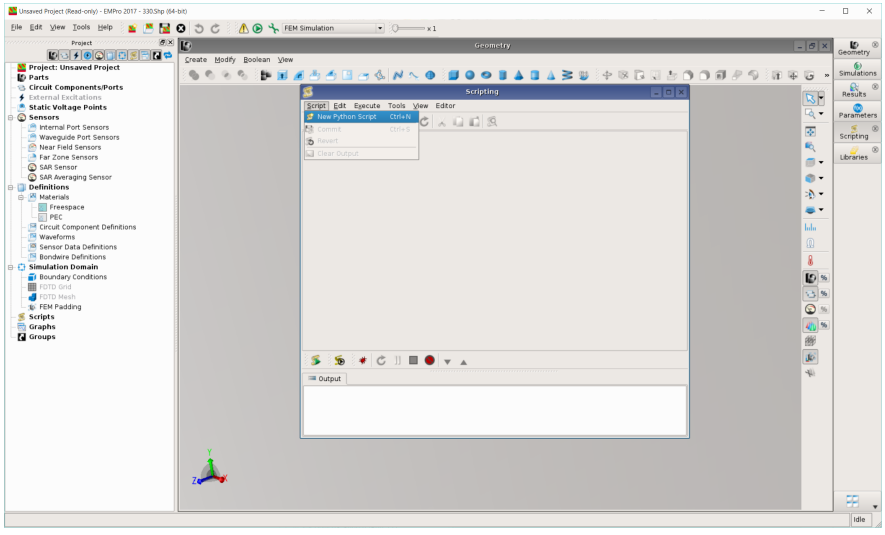


Figure 25: Creating a new script.

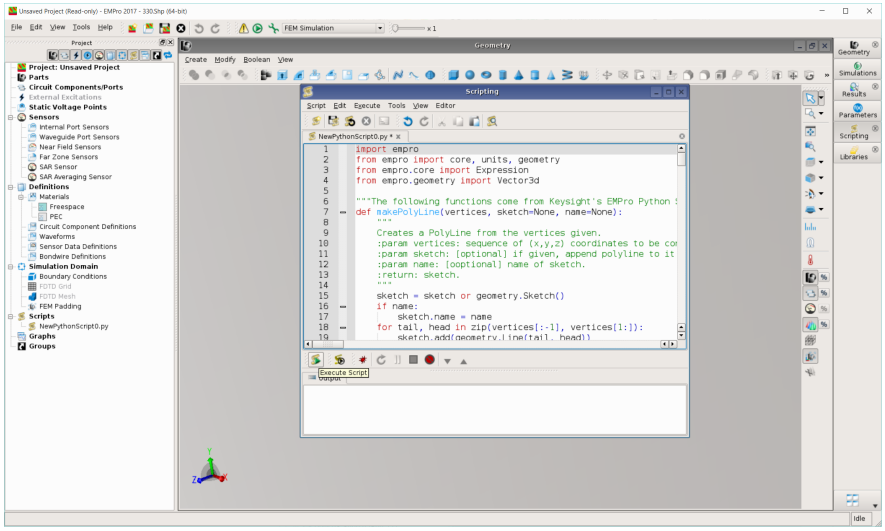


Figure 26: Executing the script.

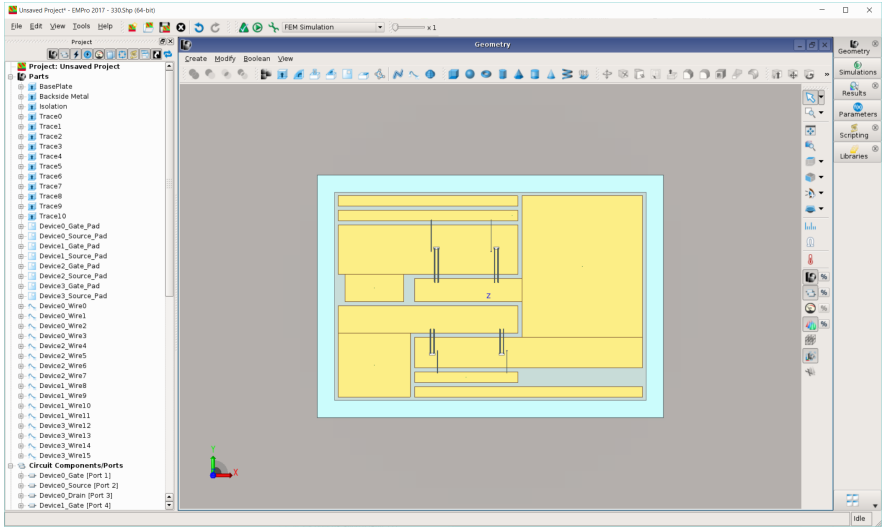


Figure 27: EMPro project creation completed.