

# Application of Multi-level Multi-domain Modeling to a Claw-pole Alternator

Birgit Knorr, Deepika Devarajan, Dingsheng Lin, Ping Zhou, and Scott Stanton  
Ansoft Corporation, Pittsburgh, PA

Copyright © 2004 Society of Automotive Engineers, Inc.

## ABSTRACT

The emergence of diverse and powerful simulation tools makes it possible to describe automotive components at several levels of abstraction. Engineers are often required to make difficult decisions about the appropriate level of model complexity to employ: whether the component needs to be described at a detailed electromagnetic level, at an intermediate circuit level, or at an abstract behavioral level. While such decisions depend on application needs, they can result in expensive design changes since incorrect simulation at any level results in prototype failure and necessary redesign. This paper describes a new multi-level modeling methodology with a detailed analysis of the tradeoffs between the accuracy of an abstraction level and the efficiency of the simulation. The procedure is illustrated using a claw-pole alternator as an example.

## INTRODUCTION

As CAD tools become increasingly powerful, initial prototyping of component models is often performed by using software modeling and computer simulations. Automotive model engineers are often faced with the difficult choice of the level of detail that needs to be included in the component description. Diverse simulation tools are currently available making it possible to capture the model description of a component at different levels of abstraction. While some choices are clarified by specific application requirements, incorrect simulation at any level will result in design failure and expensive design changes. Often the design flow through different levels of model complexity is difficult and time-consuming. To reduce costs and to ensure that prototypes are designed right the first time, it is important to understand and characterize the tradeoffs between the accuracy of an abstraction level and its effect on system performance.

In this paper, we describe a new modeling methodology for simulating automotive electrical components in which detailed electromagnetic design, efficient circuit design and high-level system design are used interchangeably. The methodology is illustrated by the modeling of an automotive claw-pole alternator. Three tools employed

from the Ansoft Design Suite – RMxp<sup>rt</sup><sup>™</sup>, Maxwell<sup>®</sup>, and SIMPLORER<sup>®</sup> provide three different levels of simulation. The claw-pole alternator design is started from RMxp<sup>rt</sup>, an electrical machine analysis tool that incorporates design specifications, physical dimensions, winding characteristics, and material properties of a specific machine type. Built-in interfaces within RMxp<sup>rt</sup> enable data transfer of the model characteristics either to a more detailed FEA (Finite Element Analysis) tool or to a state-space based tool for system simulation. We export the claw-pole alternator model geometry from RMxp<sup>rt</sup> to Maxwell 3D FEA to provide accurate field computation including non-linearity, induced eddy currents, and mechanical movement. FEA assures accurate performance evaluation since its elements can be easily adapted to any shape of boundary and interface geometry.

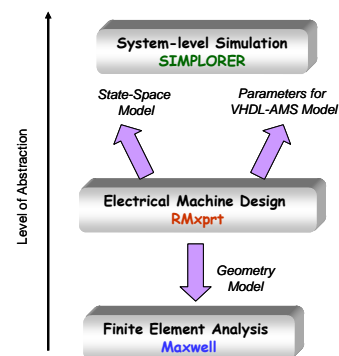


Fig. 1 Design flow for multi-level modeling

We export a state-space model of the claw-pole alternator based on a multi-dimensional non-linear table for system simulation. The state-space simulation is performed in SIMPLORER, a multi-domain simulation package combining circuit, state machine, and block diagram simulation. We also develop an abstract behavioral system-level model of the claw-pole alternator using the IEEE standardized tool-independent VHDL-AMS language capability within SIMPLORER. The parameters for the equation-based VHDL-AMS model are obtained from RMxp<sup>rt</sup>.

Fig. 1 outlines the design flow adopted for this claw-pole alternator problem. A simple automotive test circuit is developed as a schematic in SIMPLORER and the two system level models - the state-space model and the VHDL-AMS model, are analyzed. The simulation characteristics are explained in detail below with respect to the model accuracy, and minor deviations in model behaviors are analyzed.

## ELECTRICAL MACHINE DESIGN

A claw-pole alternator is a special kind of synchronous generator. The slotted stator is usually equipped with a poly-phase AC winding. The claw-pole rotor is equipped with a cylindrical permanent magnet and/or a field winding excited by DC current. Due to the complex three-dimensional structure of claw poles, the magnetic field inside a claw-pole alternator needs to be treated as a 3-dimensional (3D) problem. The specific claw-pole alternator model that we develop in this paper is a three-phase six pole-pair model that is used in an automotive application operating in a 42V powernet. The rated alternator speed is 20000 rpm and the rated power is 9KVA. The stator has 36 slots with 36 turns per phase and the field winding has 400 turns excited by a 4A DC field current.

The RMxprt tool is useful in creating a preliminary parameterized model of the claw-pole alternator based on the above specifications. The parameterized input dialogs in the software tool allow designers to input the alternator data easily, such as ratings, stator and rotor sizes, slot forms, materials, and winding arrangements, as shown in Fig. 2.

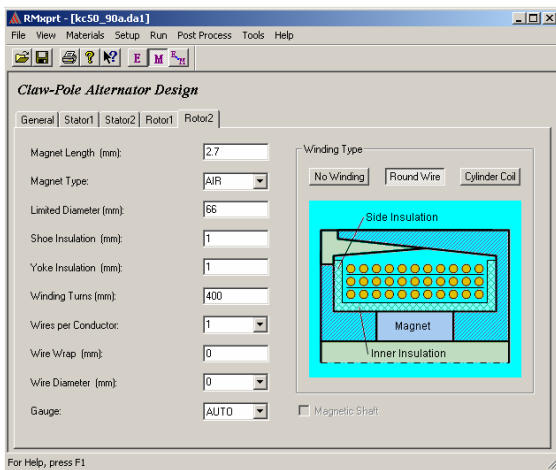


Fig. 2 Design Data Input in RMxprt

For most standard windings, the winding configuration can be automatically determined by RMxprt based on the winding type chosen by the designer. For some special windings, designers can define the winding arrangement using a winding editor, as shown in Fig. 3.

Due to the analytic nature of the design algorithms in RMxprt, results are obtained within seconds. This fast analysis allows designers to modify designs quickly and

easily to investigate different solutions in an extremely short time. After the analysis, users have a variety of design outputs choices available. A design output sheet is generated – this can be used as a specification for suppliers as well as providing lamination and winding information. Fig. 4 shows one of the available performance curves – output power vs. power angle.

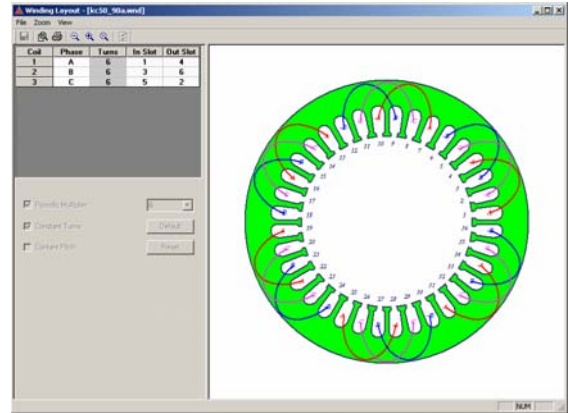


Fig. 3 Winding Editor

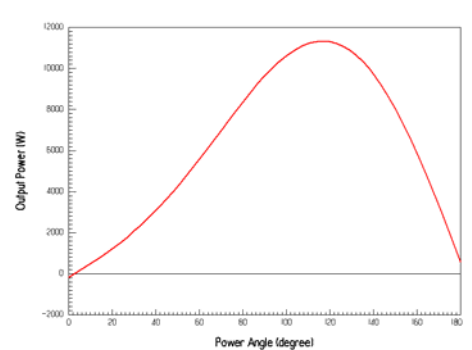


Fig. 4 Performance Curve Output

## FEA TRANSIENT ANALYSIS

The RMxprt design stage calculates performance characteristics according to the specifications and thereby narrows the design options. To obtain more detailed and accurate analyses, transient FEA must be applied to the RMxprt model. Due to the 3D nature of a claw-pole alternator, 3D FE analysis is required. FE analysis computes the behavior of the alternator precisely from first principles by taking the geometric complexity, non-linearity, induced eddy currents and mechanical movement into account simultaneously. While this approach guarantees a realistic physical model, it requires greater computational resources than is required by the analytical model in RMxprt.

RMxprt can export the claw-pole alternator geometry model directly to Maxwell 3D for FE analysis. This direct link between RMxprt and Maxwell eliminates the tedious process of creating the 3D geometry in Maxwell. After the geometry import, the user can easily set up boundary conditions, assign material properties, and describe the drive circuit and various sources.

To reduce computation costs with 3D problems, instead of treating a vector in the whole domain, we use just a magnetic scalar potential  $\Omega$  in the whole domain and a current vector potential  $T$  in the conducting region. In this case, the magnetic field  $H$  is split into four components:

$$\mathbf{H} = \mathbf{H}_s + \mathbf{T} + \nabla\Omega + \sum \mathbf{T}_{wk}$$

Here  $H_s$  is the source field due to known total currents or known current densities in either solid conductors or in stranded windings, and  $T_{wk}$  is the source field due to unknown currents in a voltage-driven stranded winding  $k$  or in a voltage-driven solid conductor loop  $k$ . The source field  $H_s$  can be any field satisfying Ampere's Law in integral form. Thus, the source field distribution can be constructed by setting the line integral of  $H_s$  on the edges of a tree of the mesh to zero and by assigning the total current enclosed by the surface of the loop to the relevant co-tree edge. For voltage-driven coils, the currents are not known. The source field  $T_{wk}$  can be expressed as

$$\mathbf{T}_{wk} = i_k \mathbf{H}_k$$

where  $H_k$  is a source field representing 1A current flowing in voltage-driven winding  $k$  and  $i_k$  is the unknown current to be determined;  $H_k$  can be considered as a basis function for the unknown source field  $T_{wk}$ . In addition, when permanent magnets are involved, the magnetic field  $H$  includes an additional term: the coercivity  $H_c$ .

The problem region is discretized into tetrahedra, the scalar field associated with  $\Omega$  is represented by node-based shape functions and the vector field is represented by edge-based shape functions. Applying Galerkin's method, the discretized field equations can be expressed in matrix form as

$$\left( [S] + [Q] \frac{d}{dt} \right) \{T\} + \frac{d}{dt} [B] \{\Omega\} + \frac{d}{dt} [H] \{i_w\} = \{g\}$$

where  $\{T\}$  and  $\{\Omega\}$  are column matrices of the edge values of the current vector potential and of the nodal values of magnetic scalar potential, respectively,  $\{i_w\}$  is a column matrix of currents in stranded windings or in solid conductors associated with a voltage source, and the matrices  $[S]$ ,  $[Q]$ ,  $[B]$ ,  $[H]$  and column matrix  $\{g\}$  arise from the standard Galerkin's method. In ferromagnetic materials, the matrices  $[Q]$ ,  $[B]$  and  $[H]$  are nonlinear.

Conventional approaches require considerable computational time to solve the multivalued problem associated with the use of scalar potential when the non-conducting region is multiply connected. To reduce this time, an automatic algorithm to generate a simply connected domain is implemented in Maxwell 3D. This is accomplished by introducing cutting domains composed of a single layer of elements to fill each hole in the conductors. The zero curl condition on the vector

potential in the cutting domains is strongly imposed by using a constraint equation for each hole.

The arbitrary movement of the rotor is handled by a moving surface method. This technique couples the two independent meshes of stationary part and the moving part along a sliding surface based on the underlying finite element shape functions. One of the advantages of this method is that remeshing is completely eliminated.

Maxwell 3D also provides a powerful post processing capability for data manipulation and visualization. Interactive visualization of detailed field quantities provides enhanced understanding of electric machine behaviors. In addition, essential engineering quantities, such as various currents and induced voltages, flux linkage, torque, speed, position, power loss and core loss are also directly available. Thus, a non-specialist is able to design a machine with confidence.

The claw-pole alternator model analyzed here is specified to have 36 stator teeth and 6 pole pairs. Since the geometry of the alternator is symmetric over two pole pitches, the minimum field computation domain can be reduced to 1/6 of the full model by using periodic boundary conditions. This provides greater computational efficiency. Fig. 5 shows the shaded plot of flux density at 20,000 rpm. An excitation current of 4A is impressed in the rotor winding. The stator three-phase windings can be treated either as voltage load or as current load based on actual operating conditions. Fig. 6 shows the eddy-current distribution in the claws.

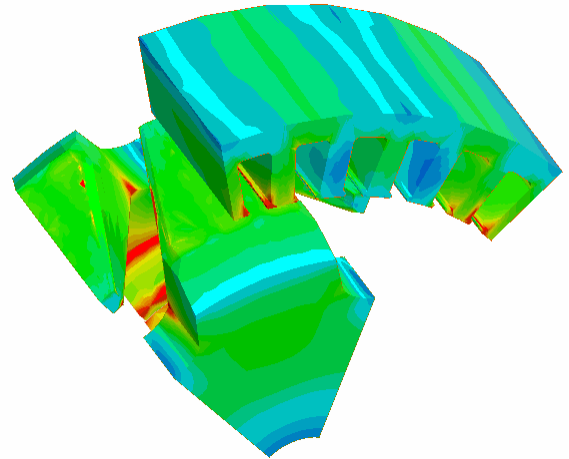


Fig. 5 Flux density plot

Since the rotor claws are press-formed out of solid metal, knowledge of the eddy-current distribution in the claws is very useful to develop an optimum design. The distribution of power loss and core loss can further be used as the thermal load for thermal analysis using the Maxwell ePhysics package. This package also provides the stress resulting from the computed thermal strain and nodal force distribution.

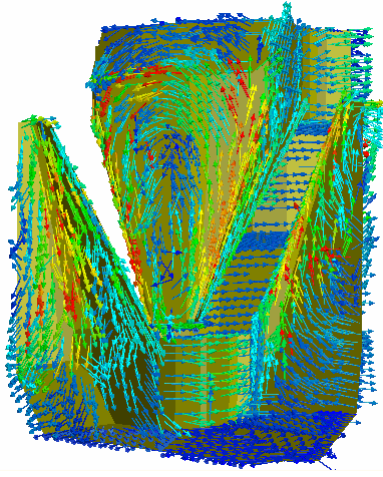


Fig. 6 Induced eddy current distribution

## STATE-SPACE MODEL

The claw-pole alternator model developed with the required specifications in RMxprt and refined with Maxwell 3D can also be exported as a state space model for system level simulation. In this case, the non-linear behavior of the alternator is represented as characteristic data defined in a multi-dimensional lookup table. This export function is a single-click operation that prompts RMxprt to create a SIMPLORER model file that may be included in a SIMPLORER project. SIMPLORER is a multi-domain simulation package combining circuit, state machine, and block diagram simulation. It is based on a unique simulator coupling technology that uses simulation algorithms that are tailor-made for different modeling domains. The state-space model can then be simulated with a test circuit to analyze the system-level behavior of the alternator model.

## VHDL-AMS MODEL

VHDL-AMS is an IEEE standardized modeling language that lends itself to mixed-signal, multi-domain modeling at different levels of abstraction. Models developed in VHDL-AMS are not simulator dependent and can be easily exchanged in plain-text format between tools from different vendors. A simple system-level equation-based math model of the claw-pole alternator may be obtained in VHDL-AMS by considering a linear model with three-phase output.

Modeling in VHDL-AMS may be viewed as a two-step process – in the first step, the interface of the model is specified in an “entity” description; in the second step, the model behavior is specified in an “architecture” description. The entity description allows the specification of both conservative and non-conservative pins/ parameters that are used in the model. In the claw-pole model, three pairs of conservative electrical pins for the three-phase output, a pair of electrical pins for the field circuit and a pair of translational (speed-torque) mechanical pins for the mechanical input are provided as *Terminal* objects. Model parameters that remain

constant during the simulation such as field inductance, armature resistance, number of pole pairs, etc. are provided as *Generic* objects in the entity description. Fig. 7 shows the entity description of the claw-pole alternator.

```

ENTITY clawpole IS
  GENERIC (
    ra : REAL := 40.0e-3; -- Armature Resistance
    ld : REAL := 42.0e-3; -- D-axis Inductance
    lq : REAL := 42.0e-3; -- Q-axis Inductance
    lfd : REAL := 35.0e-3; -- Mututal Inductance between d-axis and field
    rf : REAL := 1.0; -- Field Resistance
    lf : REAL := 50.0e-3; -- Field Inductance
    l0 : REAL := 0.0; -- Zero-sequence Inductance
    p : INTEGER := 2; -- Number of Pole Pairs
  )
  PORT (
    TERMINAL pa1,pa2,pb1,pb2,pc1,pc2 : ELECTRICAL;
    TERMINAL pf1,pf2 : ELECTRICAL;
    TERMINAL pm1,pm2 : ROTATIONAL_V;
  )
END ENTITY clawpole;

```

Fig. 7 VHDL-AMS Entity Description

The claw-pole alternator model behavior is based on the transformation of the three-phase inputs (ABC) to the DQ0 axis equations as shown in Fig. 8. The transformations work left-to-right for a motor model and right-to-left for an alternator model.

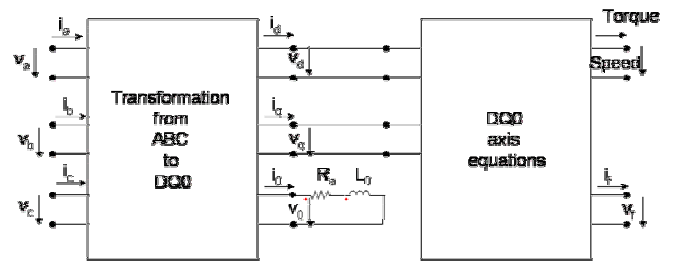


Fig. 8 Behavioral description of motor/alternator model

First, the three phase currents and voltages are transformed from ABC to DQ0 and then DQ axis equations are applied to determine the speed and torque of the rotational terminal. A zero-sequence inductance and the armature resistance are modeled as a series connection in the ‘0’ line of the DQ0 axis. The equations that specify this transformation are as follows<sup>1</sup>:

Transformation from ABC to DQ0

$$\begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos \theta & \cos(\theta - 2\pi/3) & \cos(\theta - 4\pi/3) \\ \sin \theta & \sin(\theta - 2\pi/3) & \sin(\theta - 4\pi/3) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \cdot \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix}$$

DQ Axis Equations

$$\begin{bmatrix} v_d \\ v_q \\ v_f \end{bmatrix} = \begin{bmatrix} R_a + L_d p & L_q \dot{\theta} & L_{Fd} p \\ -L_d \dot{\theta} & R_a + L_q p & -L_{Fd} \dot{\theta} \\ L_{Fd} p & 0 & R_f + L_f p \end{bmatrix} \cdot \begin{bmatrix} i_d \\ i_q \\ i_f \end{bmatrix}$$

$$\begin{bmatrix} i_d \\ i_q \\ i_0 \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos \theta & \cos(\theta - 2\pi/3) & \cos(\theta - 4\pi/3) \\ \sin \theta & \sin(\theta - 2\pi/3) & \sin(\theta - 4\pi/3) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \cdot \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$

where  $\theta_m$  and  $\theta$  are the position in mechanical and electrical degrees respectively,  $p$  stands for the operation of  $d/dt$ , and  $\dot{\theta} = d\theta/dt$ .

The output torque equation is described as:

$$T = pp \cdot [i_d i_q (L_q - L_d) - L_{Fd} i_q]$$

where  $pp$  is the number of pole pairs.

The architecture description in VHDL-AMS that implements the above equations is provided in Fig. 9. Mathematical constants that are used in the model are built using the MATH\_PI, SQRT etc definitions that are available in the MATH\_REAL VHDL-AMS package. The model uses six local terminals in the model architecture to perform the transformation between the two coordinate systems. Branch quantities (analog values) are declared between conservative terminal pins and free quantities are used to store temporary values in the model. The mechanical and electrical angles are calculated based on the angular velocity of the rotational terminals. The torque is calculated based on the currents in the DQ0 co-ordinates and the associated inductance values. VHDL-AMS allows the easy specification of differential algebraic equations through simultaneous statements that use the 'DOT (pronounced tick-dot) attribute to obtain quantity derivatives.

```

ARCHITECTURE behav OF clawpole IS
-- ***** CONSTANTS *****
CONSTANT sqrt2by3 : REAL := SQRT(2.0/3.0);
CONSTANT twopi3 : REAL := MATH_2_PI/3.0;
CONSTANT fourpi3 : REAL := MATH_2_PI*2.0/3.0;
CONSTANT onebysqrt2 : REAL := 1.0/SQRT(2.0);
CONSTANT pp : REAL := real(p);
-- ***** TERMINALS *****
-- DQ0 terminals
TERMINAL pd1,pd2,pq1,pq2,p01,p02 : ELECTRICAL;
-- ***** BRANCH QUANTITIES *****
-- Three phase terminals' quantities
QUANTITY va ACROSS ia THROUGH pa1 TO pa2;
QUANTITY vb ACROSS ib THROUGH pb1 TO pb2;
QUANTITY vc ACROSS ic THROUGH pc1 TO pc2;
-- Field voltage and current
QUANTITY vfl ACROSS ifl THROUGH pf1 TO pf2;
-- DQ0 terminals's quantities
QUANTITY vd ACROSS id1 THROUGH pd1 TO pd2;
QUANTITY vq ACROSS iq1 THROUGH pq1 TO pq2;
QUANTITY v0 ACROSS i01 THROUGH p01 TO p02;
QUANTITY vd1 ACROSS id THROUGH pd1 TO pd2;
QUANTITY vq1 ACROSS iq THROUGH pq1 TO pq2;
QUANTITY v01 ACROSS i0 THROUGH p01 TO p02;
-- Mechanical output quantities
QUANTITY vel ACROSS tor THROUGH pm1 TO pm2;
-- ***** FREE QUANTITIES *****
QUANTITY theta_m, theta_e : REAL := 0.0;

BEGIN
-- Calculation of Mechanical and Electrical Angles
theta_m == vel'integ;
theta_e == pp*theta_m;

-- Equations for transformation from ABC to DQ0 co-ordinates
vd == sqrt2by3*(cos(theta_e)*va + cos(theta_e-twopi3)*vb + cos(theta_e-fourpi3)*vc);
vq == sqrt2by3*(sin(theta_e)*va + sin(theta_e-twopi3)*vb + sin(theta_e-fourpi3)*vc);
v0 == sqrt2by3*(onebysqrt2*va + onebysqrt2*vb + onebysqrt2*vc);

id == sqrt2by3*(cos(theta_e)*ia + cos(theta_e-twopi3)*ib + cos(theta_e-fourpi3)*ic);
iq == sqrt2by3*(sin(theta_e)*ia + sin(theta_e-twopi3)*ib + sin(theta_e-fourpi3)*ic);
i0 == sqrt2by3*(onebysqrt2*ia + onebysqrt2*ib + onebysqrt2*ic);

v0 == 10*ra + 10*10'dot;

-- Equations for DQ-Axis
vd == ((ra*id)+(ld*id'dot))+ ifd*(ifl'dot);
vq == (-ld)*(theta_e'dot)*id + ((la*iq)+(lq*iq'dot))+ (-ld)*(theta_e'dot)*ifl;
vfl == (lfd)*id'dot + (tf*ifl + (ifl*i'dot));

-- Output torque equation
tor == pp*(id*iq*(lq-ld)-lfd*ifl*iq);

END ARCHITECTURE behav;

```

Fig. 9 VHDL-AMS Architecture Description

## SYSTEM-LEVEL SIMULATION

The two system-level models developed in the previous section are first evaluated using a no-load test-circuit as shown in Fig. 10.

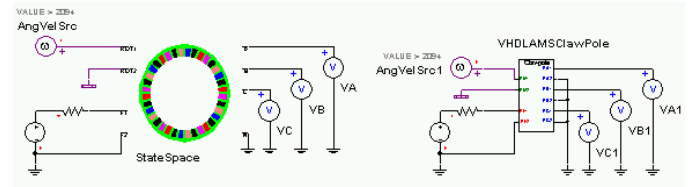


Fig. 10 No-load test-circuit

The claw-pole alternators are driven with angular velocity sources at 20,000 rpm (or 2094 rad/s) and excited with 4A field current. When the output characteristics of the no-load excitation are compared in Figure 11, it is found that the VHDL-AMS model delivers a sinusoidal output while the state-space model has a more accurate output waveform. This can be attributed to the fact that the VHDL-AMS model is a linear model and uses only the fundamental component of the EMF. On the other hand, the state-space model considers the saturation change with the load as well as the harmonics in the EMF.

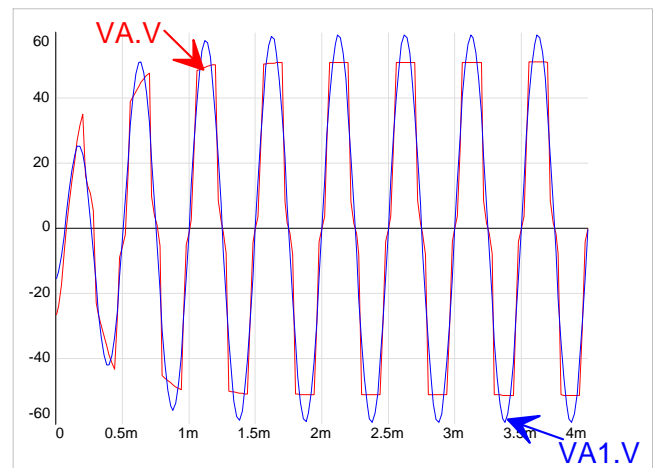


Fig. 11 No load output characteristics

Next, the state-space model is simulated in a system-level automotive test circuit as shown in Fig. 12.

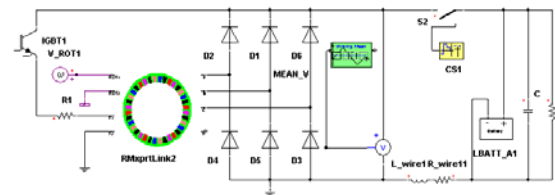


Fig. 12 State-space model in automotive test circuit

In this circuit, the state-space model of the alternator is driven at a speed of 20,000 rpm using an angular velocity source. The excitation current is delivered by the battery voltage and can be switched off by the IGBT

model. The 42V battery used is from SIMPLORER's automotive library and is a dynamic model parameterized with 18 cells. The output voltage from the alternator model is rectified by a three-phase bridge circuit. A voltmeter measures the rectified output voltage across the bridge and the mean value from this voltmeter is determined. The load is assumed as resistive for this initial simulation and is set at 10Ω. Fig. 13 shows the excitation current and Fig. 14 depicts the supplied voltage during the start phase.

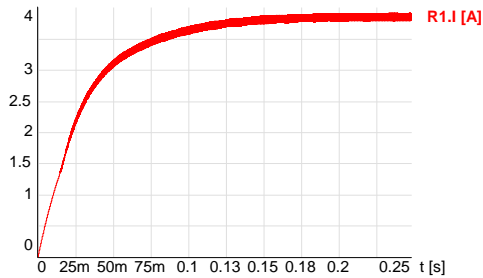


Fig. 13 Excitation Current

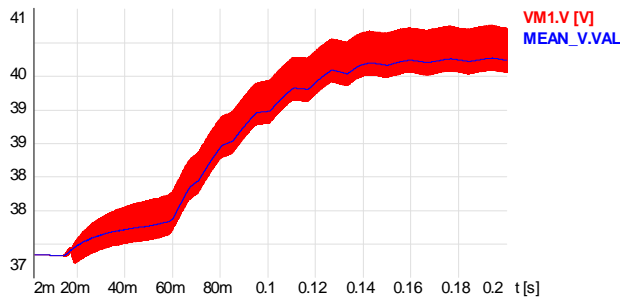


Fig. 14 Supplied Voltages

The battery provides the initial voltage that provides the excitation current for the alternator but after 60ms, the output voltage of the battery exceeds that of the alternator. Consequently, a positive current flows into the battery then and this is reflected by an increase in the battery voltage. These characteristics are visualized in Fig. 15.

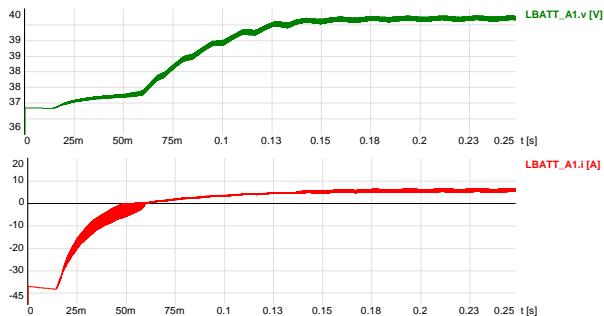


Fig. 15 Battery Voltage and Current

Finally, we analyze the loss in accuracy if we use the simple behavioral VHDL-AMS model instead of the

state-space model in the test circuit shown in Figure 10. The output voltages from the two system-level models are plotted in Fig. 16 and it is observed that the state-space model delivers a slightly smaller mean voltage (40.27V) compared to that of the VHDL-AMS model (40.30V). This difference may be attributed to the fact that the state-space model considers the harmonics in the induced voltages while the VHDL-AMS model considers only the fundamental induced voltage (refer Fig. 11). During the process of creating the state-space model and parameterization by RMxprrt, nonlinearities are considered based on the expected operating conditions.

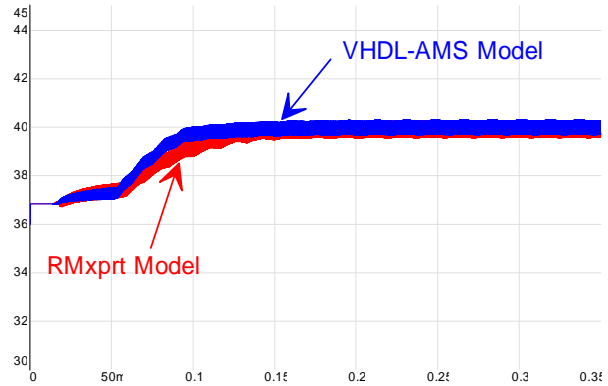


Fig. 16 Comparison of output voltages

## CONCLUSION

One of the most challenging tasks for a design engineer is to obtain the motor design parameters from design data such as physical dimensions, material properties, etc. The traditional method of developing a motor model from measurement and experiments is expensive and can take several weeks. The RMxprrt motor design tool greatly simplifies this process and not only provides the model parameters in seconds but also exports models for either FEA or system level simulation. Additionally, since models developed using the traditional method are not reusable, any changes made to the physical motor design would require another time consuming design cycle. Since RMxprrt allows quick changes in physical design parameters, it reduces time to market significantly.

To further obtain more detailed and accurate analysis, it is desirable to apply 3D time stepping transient finite element analysis to directly study transient as well as steady-state performance of a device from first physics principles. The price to be paid is it requires considerable computation time. This analysis process can be applied in the final design stage based on the narrowed design space.

Due to the large number of proprietary languages supported by simulation tools, model portability across tools from different vendors has become an increasingly important issue. VHDL-AMS has been standardized by the IEEE as a mixed-signal multi-domain modeling

language and is rapidly being accepted by the automotive industry. In this paper, we developed an analytical alternator model, parameterized it with values derived from RMxp<sub>r</sub>t, simulated it in the system-level package SIMPLORER, and compared the results with a state-space model exported from RMxp<sub>r</sub>t. While the RMxp<sub>r</sub>t model is able to characterize the alternator's non-linear behavior, the VHDL-AMS model has the advantage of model exchangeability.

As manufacturing density in automotive systems swells, thermal issues will become more important. Future research and development will include thermal and other stress effects in both FE analysis as well as state space models. Also, rapid developments in semiconductor technology offering ever-increasing computational power are now opening new avenues of research where system level simulation and FEA can be coupled for

highly accurate results. While we expect a significantly increased accuracy in simulation results from such an approach, the simulation times will still be a limiting factor.

## **ACKNOWLEDGMENTS**

All simulation models and analysis results presented in this paper were generated using Maxwell®, RMxp<sub>r</sub>t™ and SIMPLORER® products from Ansoft Corporation. Maxwell, RMxp<sub>r</sub>t and SIMPLORER are trademarks of Ansoft Corporation.

## **REFERENCE**

1. Hancock, N.N., Matrix Analysis of Electrical Machinery, 2<sup>nd</sup> Edition, Pergamon Press, 1974