August 1999 • NREL/JA-540-26839

ADVISOR 2.1: A User-Friendly Advanced Powertrain Simulation Using a Combined Backward/Forward Approach

K.B. Wipke, M.R. Cuddy, and S.D. Burch

Preprint prepared for IEEE Transactions on Vehicular Technology: Special Issues on Hybrid and Electric Vehicles

National Renewable Energy Laboratory

1617 Cole Boulevard
Golden, Colorado 80401-3393

NREL is a U.S. Department of Energy Laboratory
Operated by Midwest Research Institute • Battelle • Bechtel

Contract No. DE-AC36-98-GO10337

NOTICE

This report was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or any agency thereof.

Available to DOE and DOE contractors from:
Office of Scientific and Technical Information (OSTI)
P.O. Box 62
Oak Ridge, TN 37831
Prices available by calling 423-576-8401

Available to the public from:
National Technical Information Service (NTIS)
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
703-605-6000 or 800-553-6847
or
DOE Information Bridge
http://www.doe.gov/bridge/home.html

Printed on paper containing at least 50% wastepaper, including 20% postconsumer waste

ADVISOR 2.1: A User-Friendly Advanced Powertrain Simulation Using a Combined Backward/Forward Approach

Keith B. Wipke, Matthew R. Cuddy, and Steven D. Burch

Abstract--ADVISOR 2.1 is the latest version of the National Renewable Energy Laboratory's advanced vehicle simulator. It was first developed in 1994 to support the U.S. Department of Energy hybrid propulsion system program, and is designed to be accurate, fast, flexible, easily sharable, and easy to use. This paper presents the model, focusing on its combination of forward- and backward-facing simulation approaches, and evaluates the model in terms of its design goals. ADVISOR predicts acceleration time to within 0.7% and energy use on the demanding US06 to within 0.6% for an underpowered series hybrid vehicle (0-100 km/h in 20 s). ADVISOR simulates vehicle performance on standard driving cycles between 2.6 and 8.0 times faster than a representative forward-facing vehicle model. Due in large part to ADVISOR's powerful graphical user interface and Web presence, over 800 users have downloaded ADVISOR from 45 different countries. Many of these users have contributed their own component data to the ADVISOR library.

I. Introduction

ADVISOR was first developed in November 1994. Its main purpose was to help manage the U.S. Department of Energy's (DOE) hybrid electric vehicle (HEV) program subcontracts by facilitating our understanding of the technical challenges inherent in the design of high-efficiency HEVs. ADVISOR uses drivetrain component performance to estimate fuel economy and emissions on given cycles as well as maximum-effort acceleration capability. It is fundamentally an empirical model.

A. Model design

In accordance with ADVISOR's mission as an analysis tool to support the U.S. DOE hybrid program, we designed ADVISOR to meet certain goals. It needed to be

- Accurate, allowing meaningful comparison of different drivetrain configurations
- Fast, allowing high-speed analysis of vehicles and design space investigations, such as multi-dimensional parametric studies and optimization
- Flexible, allowing us to evaluate vehicles with various control strategies and combinations of components
- Publicly available, allowing us to share it with potential collaborators and also to foster HEV development and understanding among the public
- Capable of modeling vehicles of any type: conventional, electric, series hybrid, or parallel hybrid
- Easy to use, even for those without detailed knowledge of vehicle modeling.

Vehicle simulators existing in 1994 were considered for use before ADVISOR was developed [1-3]. Existing simulators were available to NREL only as executable code. Lack of access to the source code prevented the implementation of new, unique control strategies and new vehicle configurations with these tools. Also, existing codes were not designed to fully simulate parallel HEVs or conventional-drivetrain vehicles.

To best meet our design goals, we chose to develop a hybrid backward/forward-facing vehicle simulator in the MATLAB®/Simulink® environment. MATLAB®/ Simulink® was chosen for its nearly self-documenting graphical programming environment and its wide acceptance by researchers in academia and industry. ADVISOR's hybrid modeling approach was chosen for its combination of the qualities of the two approaches: high-speed execution and good prediction of maximum effort accelerations (and other component-limited conditions). For reference, the two main vehicle simulation approaches are described below, followed by an in-depth discussion of NREL's hybrid approach.

1) Generic backward-facing approach

Vehicle simulators using a backward-facing approach answer the question "Assuming the vehicle met the required trace, how must each component perform?" No model of driver behavior is required in such models. Instead, the force required to accelerate the vehicle through the time step is calculated directly from the required speed trace. The required force is then translated into a torque (often by assuming some efficiency) that must be provided by the component directly upstream, and the vehicle's linear speed is likewise translated into a required rotational speed. Component by component, this calculation approach carries backward through the drivetrain, against the tractive power flow direction, until the fuel use or electrical energy use that would be necessary to meet the trace is computed.

The main author is with the National Renewable Energy Laboratory, Golden, CO 80401 USA.

The backward-facing approach is convenient because automotive drivetrain components tend to be tested so that a table of efficiency or loss versus output torque and speed is developed. This means that a straightforward calculation can determine a component's efficiency and allow the calculation to progress. The explicit nature of the efficiency/loss calculation also allows very simple integration routines (i.e., Euler) to be used with relatively large time steps on the order of 1 s. Thus, simulations using the backward-facing approach tend to execute quickly.

Weaknesses of the backward-facing approach come from its assumption that the trace is met and from the use of efficiency or loss maps. Because the backward-facing approach assumes that the trace is met, this approach is not well-suited to computing best-effort performance, such as occurs when the accelerations of the speed trace exceed the capabilities of the drivetrain. Also, because efficiency maps are generally produced by steady-state testing, dynamic effects are not included in the maps or in the backward-facing model's estimate of energy use. A related limitation of the backward-facing model is that it does not deal in the quantities measurable in a vehicle. For example, control signals like throttle and brake position are absent from the model, further hindering dynamic system simulation and control system development.

2) *Forward-facing approach*

Vehicle simulators that use a forward-facing approach include a driver model, which considers the required speed and the present speed to develop appropriate throttle and brake commands (often through a PI controller). The throttle command is then translated into a torque provided by the engine (and/or motor) and an energy use rate. The torque provided by the engine is input to the transmission model, which transforms the torque according to the transmission's efficiency and gear ratio. In turn, the computed torque is passed forward through the drivetrain, in the direction of the physical power flow in the vehicle, until it results in a tractive force at the tire/road interface. The resultant acceleration is computed from $a = F / m_{eff}$, where m_{eff} includes the effect of rotational inertias in the drivetrain.

The forward-facing approach is particularly desirable for hardware development and detailed control simulation. Because forward-facing models deal in quantities measurable in a physical drivetrain such as control signals and true torques (not torque 'requirements'), vehicle controllers can be developed and tested effectively in simulations. Also, dynamic models can be included naturally in a forward-facing vehicle model. Finally, the forward-facing approach is well-suited to the calculation of maximum effort accelerations, as they are essentially wide-open throttle events.

The major weakness of the forward-facing approach is its simulation speed. Drivetrain power calculations rely on the vehicle states, including drivetrain component speeds that are computed by integration. Therefore, higher-order integration schemes using relatively small time steps are necessary to provide stable and accurate simulation results.

In the following section, ADVISOR's backward-facing and forward-facing elements are described, focusing on the relationship between the two. Next, ADVISOR (including its relatively new graphical user interface [GUI]), is evaluated in the context of the design goals.

II. NREL'S BACKWARD/FORWARD DRIVETRAIN MODEL

ADVISOR uses a hybrid backward/forward approach that is closely related to the strictly backward-facing approach discussed above. ADVISOR's approach is unique in the way it handles the component performance limits in its backward-facing stream of calculations and in the addition of a simple forward-facing stream of calculations. The two overriding assumptions that describe ADVISOR's combination of the backward-facing and forward-facing approaches are:

1. No drivetrain component will require more torque or power from its upstream neighbor than it can use.
2. A component is as efficient in the forward-facing calculations as it was computed to be in the backward-facing calculations.

The role of these assumptions is highlighted in the discussion of ADVISOR's simulation approach below.

Figure 1 shows the top level of ADVISOR's series HEV model, programmed in the MATLAB®/Simulink® environment. Arrows indicate data flow; boxes represent data processing elements or groups thereof. For example, the box labeled "gearbox" contains all data processing elements, such as "Sum" and "Product" blocks and look-up tables, necessary to model the vehicle's single- or multi-speed gearbox. Arrows feeding data from left to right, such as the arrow going from the "motor/controller" block to the "power bus" block, are generally part of the backward-facing part of the model, passing torque, speed, and power requirements up the drivetrain. The arrows that loop back to pass data from right to left, such as the arrow from the "motor/controller" to the "gearbox" block, are part of the forward-facing part of the model, transmitting available torque, force, speed, and electrical power through the drivetrain. Each block references MATLAB® data, such as a loss or efficiency table, that describes the performance of the appropriate component.

To illustrate the way ADVISOR's backward- and forward-facing parts relate to each other, we consider the simulation of a hypothetical series HEV's maximum effort acceleration using ADVISOR 2.1. This will make an interesting and appropriate example because ADVISOR is unique in the way it handles drivetrain performance limits, and the drivetrain will always be operating at its limit during the maximum-effort acceleration. ADVISOR describes a maximum-effort acceleration by a 322-km/h

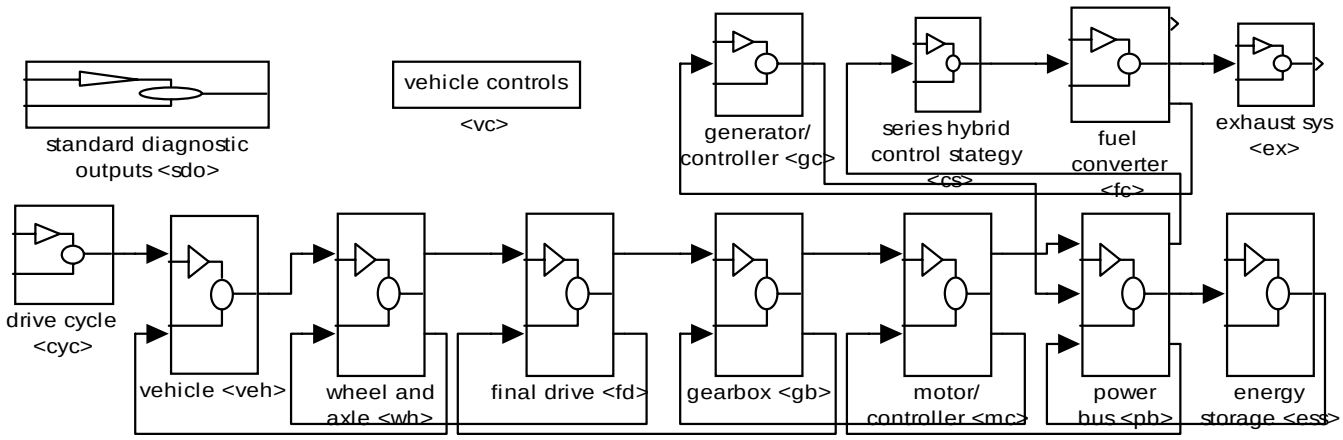

Figure 1. ADVISOR's series HEV block diagram's top level

step, assuming that this is a greater speed than the vehicle will ever reach. Below, we step through ADVISOR's calculation paths—first the 'required' values of the variables (backward-facing results) and then the 'available' values (forward-facing results).

A. Backward-facing calculation path

The leftmost block in Figure 1's chain of drivetrain components is labeled "drive cycle." This is the point at which the required speed versus time trace data is input to the simulation. The vehicle and component data defined by text files in the database are referenced in the appropriate component model. For example, all motor performance data are referenced in the "motor/controller" block.

The "drive cycle" block transmits the required speed trace to the "vehicle" block. The "vehicle" block includes no drivetrain performance limits, and straightforwardly uses the required trace to compute the average tractive force and average speed required over the time step. These requirements are passed from the "vehicle" block to the "wheel/axle" block via the lead that connects the two in Figure 1.

The "wheel/axle" block includes the transformation of force and linear speed to torque and rotational speed and the effects of tire slip, wheel and axle bearing drag, and wheel and axle rotational inertia. Only the tire slip model includes performance limits and therefore merits further discussion.

The tire slip model relates weight on the tire, longitudinal force, vehicle speed, and slip in an equation or set of tables, where

$$slip = \omega_{wh,req} \times r_{wh} / v_{req} - 1 \,. \qquad \text{(EQ. 1)}$$

(A complete list of symbols is included at the end of the paper.) The current model uses a fairly simple relationship that neglects the effect of vehicle speed. However, a model under development implements the full "magic tire" equation [4,5], which would include this effect.

The tire slip is limited to some maximum value, and this in turn limits the transmissible tractive force. Using vehicle loss parameter information borrowed from the "vehicle" block, the required speed is limited according to the acceleration possible with the traction-limited force. ADVISOR solves the following equations simultaneously at the maximum slip condition to determine the maximum force and acceleration:

$$dv/dt = F/m \qquad \text{(EQ. 2)}$$

$$F = F_{traction}(dv/dt) - F_{aero}(v) - F_{rolling}(v) - F_{climbing} \,. \qquad \text{(EQ. 3)}$$

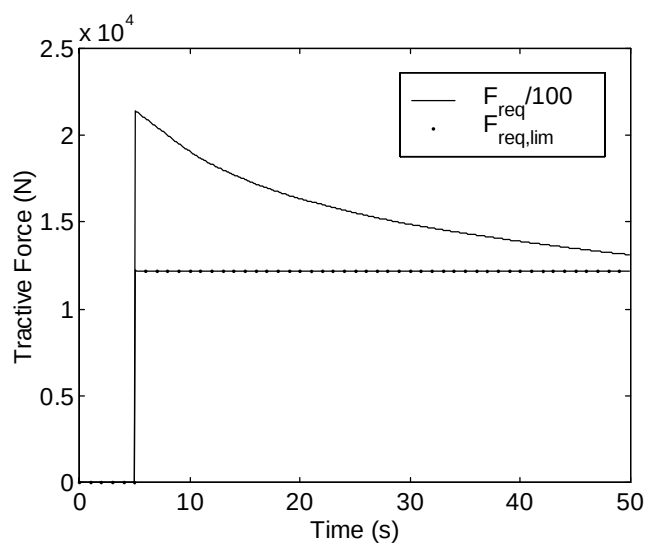

Figure 2. Required tractive force: F_{req}, required to meet trace; $F_{req,lim}$, subject to tire traction limits

As shown in Figure 2, the tractive force required to meet the trace peaks at roughly 2.1×10^6 N, coincident with the step in the trace speed, and then falls off as the vehicle

accelerates to approach the trace. (Figure 3 shows the calculated vehicle speed.) The maximum tractive force the tires can transmit is constant at roughly 1.2×10^4 N.

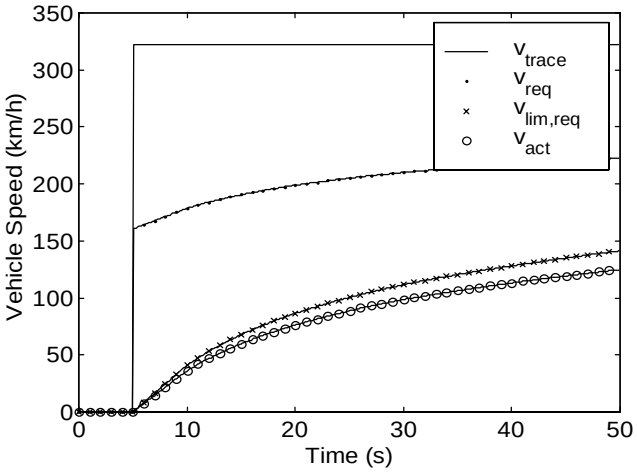

Figure 3. Required and actual vehicle speed

Figure 3 shows the various required vehicle speeds in the model. The required trace that is output by the "drive cycle" block is shown as v_{trace}. The average speed required over the time step that is output by the "vehicle" block is shown as v_{req}. The influence of the tire slip model can be seen by comparing v_{req} with $v_{lim,req}$, which is the vehicle speed possible given the tire's traction limit. Finally, v_{act} is the vehicle's actual speed, shown here for reference. Note that the actual vehicle speed is lower than the tire limit because in this example it is limited by the components "upstream" of it.

With tire slip limits enforced, the required wheel speed is calculated as follows:

$$\omega_{wh,req} = (1 + slip) v_{lim,req} / r_{wh} . \qquad \text{(EQ. 4)}$$

Required torque input to the axle is computed by summing the torque required to provide the necessary average tractive force, the torque required to overcome bearing losses and brake drag, and the torque required to accelerate the wheels' and axles' rotational inertia.

$$\tau_{wh,req} = F_{lim,req} r_{wh} + \tau_{wh,loss} + J_{wh} \left(\frac{\Delta \omega_{wh,req}}{\Delta t} \right) \qquad \text{(EQ. 5)}$$

The "wheel/axle" block sends its torque and speed requirements to the "final drive" block, which includes no limits and straightforwardly transforms the torque and speed requirements with its gear ratio and torque loss. The next in line in Figure 1, the "gearbox" block, likewise includes no performance limits. After transforming the torque and speed required of it, the "gearbox" passes the requirements upstream to the "motor/controller" block.

The next section will focus on the motor and motor controller model because it enforces a number of performance limits, and is perhaps the component model most representative of ADVISOR's hybrid backward/forward approach. Although the motor is not the end of the line of backward-facing calculations in ADVISOR, it will be the most 'upstream' component discussed in this paper. Discussion of the components further upstream such as the energy storage system would not significantly further illuminate ADVISOR's unique approach.

B. Details of motor and motor controller

The top half of ADVISOR's motor/controller model, shown in Figure 4 in the dashed box, is dedicated to the backward-facing part of the simulation. The required output torque and speed are input at the top left-hand corner of the block diagram, and the required input power is output at the top right-hand corner.

Three different performance limits are enforced in the backward-facing part of the "motor/controller" block. The required speed is limited to the motor's maximum speed. The required torque is limited to the difference between the motor's maximum torque at the limited speed and the torque required to overcome the rotor inertia. The limited torque and speed are then used to interpolate in the motor/controller's input power map. Finally, the interpolated input power is limited by the motor controller's maximum current limit. This behavior is described in the following equations:

$$P_{mot,in,req} = \min\left(P_{mot,in,map}, I_{con,max} V_{bus,prev}\right) \qquad \text{(EQ. 6)}$$

where

$$P_{mot,in,map} = f\left(\tau_{mot,lim,req}, \omega_{mot,lim,req}\right), \qquad \text{(EQ. 7)}$$

f is the functional relationship described by the motor map,

$$\omega_{mot,lim,req} = \min\left(\omega_{mot,req}, \omega_{mot,max}\right), \qquad \text{(EQ. 8)}$$

and

$$\tau_{mot,lim,req} = \min\left(f_1\left(\omega_{mot,lim,req}\right), \tau_{mot,req} + J_{mot} \left(\frac{\Delta \omega_{mot,lim,req}}{\Delta t} \right) \right) \qquad \text{(EQ. 9)}$$

where f_1 is the functional relationship described by the motor's torque envelope. For cases where the vehicle missed the required trace by more than 1.6 km/h in the previous time step, $\omega_{mot,lim,req}$ is replaced in Equations 7 and 8 and in the f_1 function evaluation in Equation 9 by the previous time step's actual motor speed, given by Equation 10.

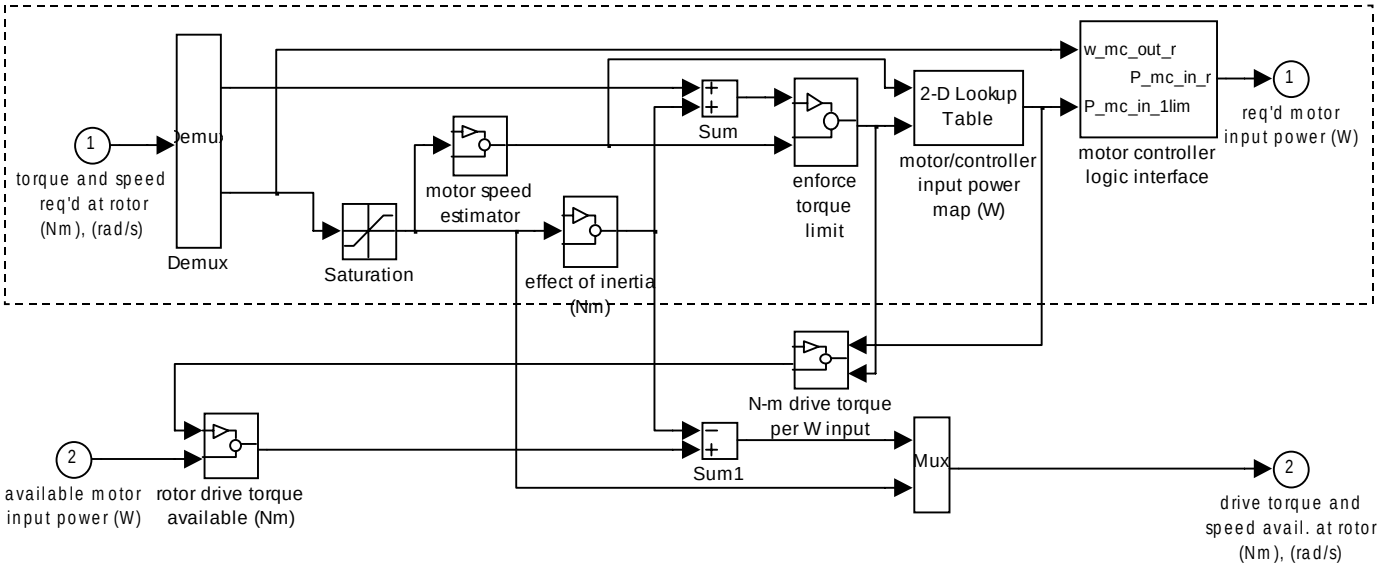

Figure 4. ADVISOR's motor/controller block diagram

$$\omega_{mot,act,prev} = \left(v_{act} \left(\frac{\omega_{mot,lim,req}}{v_{avail}} \right) \right)_{prev} \qquad \text{(EQ. 10)}$$

Figure 5 illustrates the effect of Equation 9. In the maximum-effort acceleration example, the motor is asked to produce more than its maximum torque. At times after 5 s, the maximum torque capability represented by $\tau_{mot,lim,req}$ is used to formulate the motor/controller input power requirement.

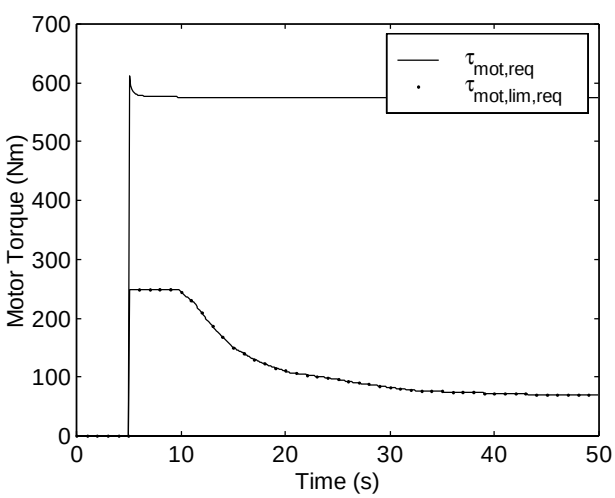

Figure 5. Required motor torque: $\tau_{mot,req}$, required into gear reduction; $\tau_{mot,lim,req}$, subject to motor torque limit

Figure 6 illustrates the effect of Equation 8. After about 42 s, the motor is required to exceed its maximum speed to provide the wheel and axle (via a gear reduction) the maximum speed they are capable of handling. $\omega_{mot,lim,req}$, coincident with $\omega_{mot,req}$ curve for most of the acceleration, is used to formulate the motor input power requirement.

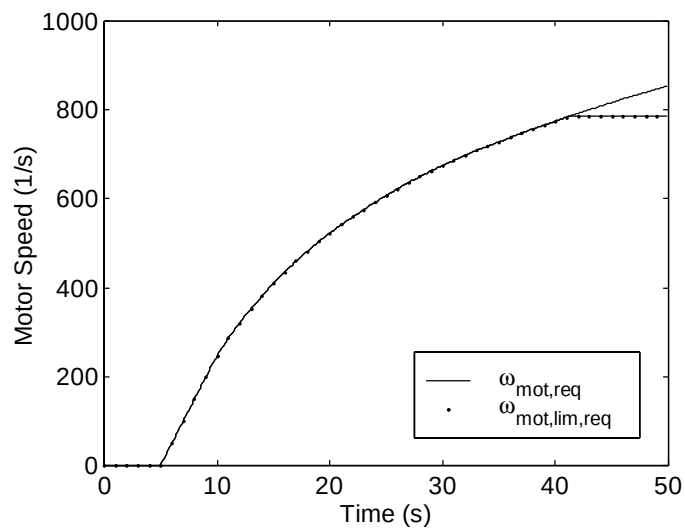

Figure 6. Required motor speed: $\omega_{mot,req}$, required into gear reduction; $\omega_{mot,lim,req}$, subject to motor speed limit

Figure 7 illustrates the effect of Equation 6. $P_{mot,in,map}$ is the input power required to power the motor at its maximum-limited torque and speed. $P_{mot,in,req}$ is the power that the motor/controller requires of the power bus, which must in turn be provided by the batteries and/or the generator. For the example case of a maximum effort acceleration, Figure 7 indicates that between about 9 s and 18 s the motor requires more power than it is capable of handling, according to its current limit, to meet the limited torque and speed requirements.

The bottom part of Figure 4, not enclosed in a dashed box, is the forward-facing part of the motor/controller model. It accepts as input the available input power, on the bottom left of the figure, and produces as outputs the available rotor torque and speed.

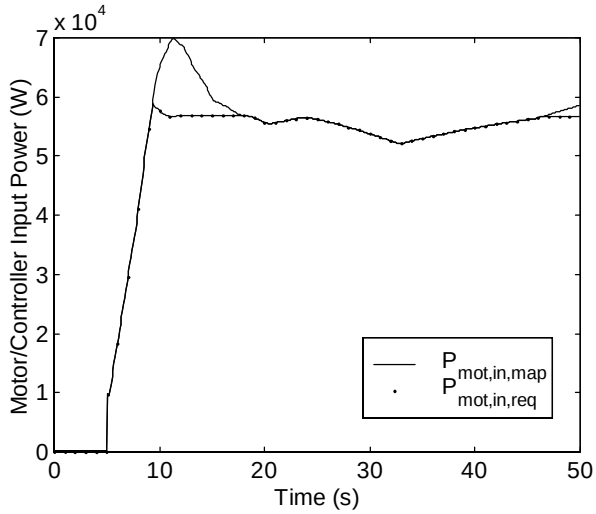

Figure 7. Required motor/controller input power: $P_{mot,in,map}$, computed by map; $P_{mot,in,req}$, subject to controller current limit

To compute the torque that can be produced by the motor/controller given the available input power, the motor/controller efficiency computed during the backward-facing calculations is used, modeled as $\tau_{mot,lim,req}/P_{mot,lim,req}$ in Equation 11 below.

$$\tau_{mot,avail} = \tau_{mot,lim,req}\left(\frac{P_{mot,avail}}{P_{mot,lim,req}}\right) - J_{mot}\left(\frac{\Delta\omega_{mot,lim,req}}{\Delta t}\right) \quad \text{(EQ. 11)}$$

Note that the model accounts for the torque required to accelerate the motor's rotor using the motor shaft's *required* acceleration. For maximum-effort acceleration runs, the required acceleration is limited by the tire slip, and this acceleration is usually greater than what is possible given the drivetrain limits. Therefore, the motor's inertial effect is overestimated for maximum-effort acceleration runs. We see below that this overestimation has negligible effect on ADVISOR's fidelity.

The motor speed that the "motor/controller" block outputs, which is termed "available" speed, is the motor's actual speed only if there are no torque or power limits active during the current time step. Figure 6 indicates that the motor model's output available speed is equal to $\omega_{mot.lim,req}$ as computed in Equation 8. This means that the "available" motor speed is the required motor speed subject to the motor's speed limit. If the available motor torque is less than the required motor torque, however, there is insufficient torque for the motor to accelerate to its required speed. This would cause the "available" motor speed output by the "motor/controller" block to be greater than the actual speed of the motor.

C. Forward-facing calculation path

The available motor torque is transformed by the efficiencies of the gearbox and the final drive (which are computed during the backward-facing calculations), and their gear reductions. This results in available drivetrain torque and speed input to the wheel and axle. Wheel slip plays a role in transforming the available speed only if it is different from the required speed, as is the case if a drivetrain component's speed limit is encountered. This is described in the equation below:

$$v_{avail} = \omega_{wh,avail} r_{wh} / (1 + slip), \quad \text{(EQ. 12)}$$

where *slip* is recomputed here using the available tractive force and v_{avail} is the component speed capability-limited vehicle speed.

Slip plays no role in computing tractive force beyond limiting the request in the backward-facing calculations. Because no calculation in the upstream components acts to increase the tractive force, the limit enforced by the slip model remains in place through the forward-facing calculations in the "wheel/axle" block.

After accounting for losses in the axle and dividing by the tire's rolling radius, ADVISOR arrives at an available tractive force. Solving Equation 3 for the speed at the end of the time step, ADVISOR arrives at an estimate of the actual vehicle speed. ADVISOR compares this force-based estimate of vehicle speed with that derived from Equation 12, and chooses the minimum of the two for the actual vehicle speed, v_{act}, plotted in Figure 3. In this way, the computed vehicle speed never exceeds that possible given the torque and force available from the drivetrain or the speed that corresponds to any drivetrain component speed limits that might be active.

III. Evaluation of the Model

A. Accuracy

Having illustrated the mathematical background of the powertrain model, we can now examine the validity and effect of the crucial assumption in ADVISOR's simplified forward-facing approach—that is, that actual component efficiency is closely approximated by that computed in the backward-facing calculations. This assumption is active only when component performance limits are encountered—when they are not, ADVISOR operates exactly as a strictly backward-facing model. We evaluate the effects of the crucial assumption by comparing ADVISOR's predictions for a performance-limited case (where the achieved speed falls short of the trace) to those for the case where the required trace is equal to the actual vehicle speed. We consider acceleration time and energy use predictions separately.

1) Simulation parameters

Range-extender series hybrid vehicles sized to achieve 0-100 km/h accelerations in 15, 20, and 25 s were simulated in this test. Their parameters are listed in Table 1.

Table 1. Test vehicle parameters

Parameter	Value
Vehicle data	
Test mass	1459, 1936, 2393 kg
Coeff. of aero. drag	0.335
Frontal area	2.0 m^2
Coeff. of rolling resistance	0.009
Motor/controller set	
Type	Permanent magnet
Maximum power	53 kW
Maximum torque	248 Nm
Maximum speed	7500 rpm
Rotor inertia	0.047 kg-m^2
Energy storage system	
Type	NiMH batteries
Pack voltage	380 V
Peak power	100 kW
Energy capacity	35 kWh
Engine/generator set	
Engine type	Spark ignition
Maximum power	41 kW
Generator type	Permanent magnet
Efficiency at operating point	95%

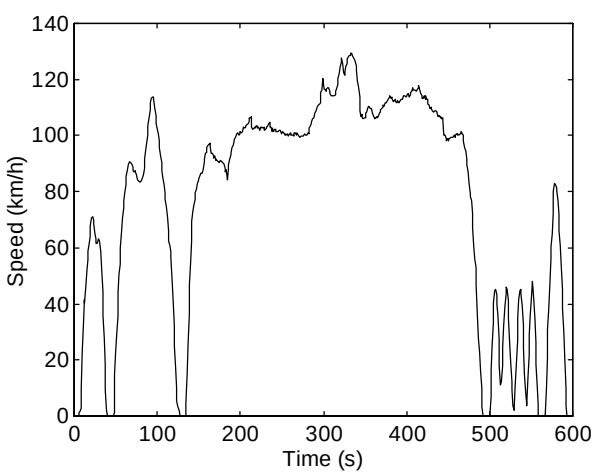

Figure 8. US06 high-speed, high-acceleration driving cycle

Two separate tests were run: one to evaluate ADVISOR's acceleration time predictions, and the other to evaluate ADVISOR's energy use predictions. For the first test, ADVISOR's hybrid approach was tested by requiring a 322-km/h step, as described in Section II above. The acceleration-time benchmarks were computed by simulating the vehicles' performance on speed traces that were iteratively defined so that they are exactly the fastest the vehicle could accelerate. In these benchmark cases, the required speed trace and the actual vehicle speed trace were the same, no component performance limits were encountered, and ADVISOR's hybrid backward/forward approach was simplified—ADVISOR became a strictly backward-facing model.

For the second test, ADVISOR's backward/forward approach was tested by simulating the vehicles' best-effort performance on the US06, the most demanding driving cycle used for vehicle certification in the United States. Even for the fastest of the three vehicles simulated here, which is nonetheless a fairly slow vehicle, the US06 is too demanding to meet the trace. The energy-use benchmarks are computed by simulating the vehicles on iteratively defined versions of the US06 so that the required speed again exactly matches the vehicles' best-effort performance.

2) Performance predictions

Table 2. 0-100 km/h acceleration times (s)

Vehicle mass (kg)	ADVISOR	Iteration	Difference
1459	15.00	15.07	-0.5%
1936	20.00	20.12	-0.6%
2393	25.00	25.19	-0.8%

Table 2 indicates that ADVISOR slightly underestimates acceleration times relative to the best estimate from iteration. ADVISOR's error is greater for grossly underpowered vehicles than for higher-powered vehicles. The computed speed traces for the three vehicles are shown in Figure 9.

Table 3. Energy use predictions (kWh)

Vehicle mass (kg)	ADVISOR	Iteration	Difference
1459	2.47	2.47	0.02%
1936	2.79	2.81	-0.7%
2393	3.06	3.12	-1.9%

Table 3 indicates that ADVISOR slightly underestimates gross electrical energy input to the motor/controller on the US06 when the simulated vehicle encounters drivetrain performance limits. These limits cause the vehicle to fall from the trace, as shown in Figure 10. The more a vehicle misses the trace, the more ADVISOR underestimates its energy use. Note that for both the energy use predictions and the acceleration time predictions, the small error introduced by the hybrid method is small enough to be

neglected relative to the effect of uncertainty in input data used to define the simulated vehicle.

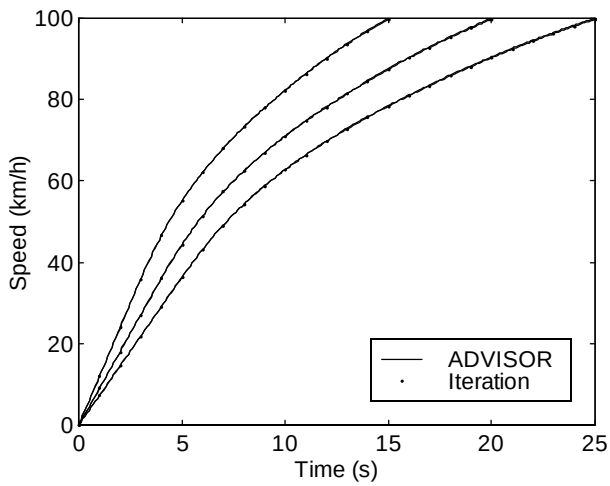

Figure 9. Maximum-effort acceleration results for the three series hybrids

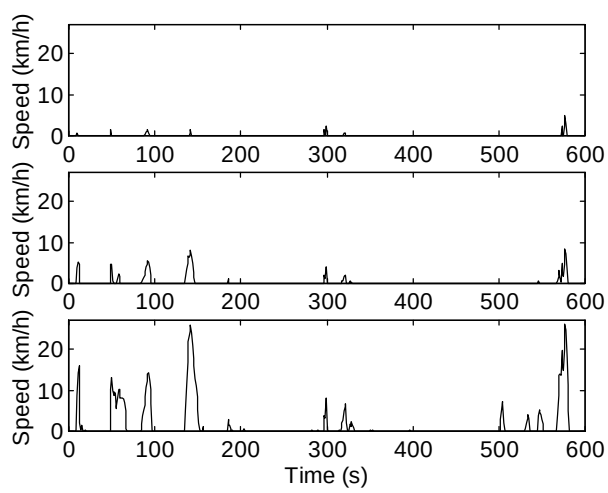

Figure 10. US06 speed trace deviation, by fastest (top graph), mid-speed (middle), and slowest (bottom) series hybrids

The results above were derived using a motor data set that includes nonzero inertia, as indicated in Table 1. By setting this inertia to zero in the model and rerunning the analyses that produced the results in Tables 2 and 3, we can estimate the effect of motor inertia on the fidelity of the model. Recall that the iteration method produces correct results (that is, consistent with a backward-facing model's predictions) regardless of the value of the motor inertia because the iteration is performed to ensure that no performance limits are encountered.

A comparison of Table 4 to the 1936-kg vehicle's results in Tables 2 and 3 indicates that the presence or absence of motor inertia does not significantly affect ADVISOR's agreement with the iteration-derived best estimate.

Table 4. Results for 1936-kg vehicle with zero motor inertia

Performance measure	ADVISOR	Iteration	Diff.
0-100 km/h time (s)	19.76	19.90	-0.7%
Energy input to motor (kWh)	2.79	2.81	-0.7%

3) Past validation and benchmarking

Exercises such as that documented above are instructive and helpful in confirming ADVISOR's behavior, but not sufficient to instill confidence that the model is consistent with other models or real test vehicles. The validation of ADVISOR to ensure its accuracy has been a high priority since its initial development. Beginning in 1995, NREL participated with representatives from industry and other national labs in a model benchmarking exercise. When all participants used identical inputs, we found that ADVISOR's predictions closely matched those of industry. When the PNGV Systems Analysis Toolkit (PNGVSAT) version 1.7 became available in April 1997, a benchmarking with that model confirmed similar results from both models for the cases studied. NREL is currently undergoing a benchmarking with a beta version PNGVSAT 2.1.

In 1997, researchers at Virginia Polytechnic Institute validated ADVISOR using data from their award-winning FutureCar competition series hybrid entry. The researchers developed data files representing their vehicle and each of its components and modified the default control strategy to match their own. They then simulated the vehicle's performance on the vehicle's actual speed trace, and compared the ADVISOR-predicted fuel-use and battery energy-use with the measured values. They found agreement within the uncertainty of their measurements [6].

B. Speed

As mentioned above, backward-facing models tend to be faster than forward-facing models largely because of the way they estimate drivetrain component speeds. Backward-facing models compute the speeds directly from the required vehicle speed trace, whereas forward-facing models integrate $a = F / m$ to compute speeds. This approach requires the use of higher order integration schemes and smaller time steps.

ADVISOR handles drivetrain speeds much in the same way that strictly backward-facing models do, and is therefore significantly faster than typical forward-facing models. Table 5 compares ADVISOR's execution times to those of a proprietary Simulink-based forward-facing model. ADVISOR and the forward-facing model were both used to simulate a conventional-drivetrain vehicle, and were run on the same 200-MHz Intel® Pentium Pro-equipped personal computer running Microsoft® Windows 95.

Table 5. Execution times for standard analysis tasks (seconds)

Test	ADVISOR	Forward-facing model	ADVISOR faster by X times
max. accel.	16	28	1.7X
FUDS	44	352	8.0X
HFET	32	83	2.6X

The FUDS (Federal Urban Driving Schedule) is also known as the first 1372 s of the U.S. EPA's Federal Test Procedure. It reaches a maximum speed of 91.2 km/h and has an average speed of 31.4 km/h. The HFET is the U.S. EPA's Highway Fuel Economy Test, which lasts 765 s, reaches a maximum speed of 96.4 km/h, and has an average speed of 77.6 km/h.

The table indicates that ADVISOR runs these tests 1.7 to 8 times faster than the forward-facing model. ADVISOR is slowest relative to the forward-facing model for the combined test including the maximum effort acceleration because ADVISOR takes 0.1-s steps for the acceleration as opposed to the 1-s steps it takes to follow driving cycles.

C. *Flexibility*

ADVISOR's flexibility must be evaluated on two levels. On one level, ADVISOR may be modified through the GUI. There are conventional-drivetrain vehicle, electric vehicle, series HEV, and parallel HEV block diagrams that may be selected through the GUI. The GUI also allows easy access to a library of 85 component and control logic files that may be used interchangeably. On the other hand, the GUI does not allow the user to develop entirely new drivetrain architectures, such as one for a parallel hybrid where the electric motor is on the wheel side of the multi-speed transmission rather than the engine side.

On another level, ADVISOR may be modified at the block diagram level, by programming in Simulink®. The ADVISOR file set that may be downloaded from the Web includes all elements of the source code. As such, ADVISOR may be treated as a toolbox of files and component models that may be connected in any number of ways. Because ADVISOR relies heavily on the backward-facing approach for its operation, its drivetrain model does not represent the drivetrain as directly and intuitively as a forward-facing model does. This may tend to complicate the disconnecting and reconnecting of block diagrams to model new vehicle types. Nonetheless, such modifications are quite possible. For example, a researcher in Germany developed a model of a four wheel-drive split parallel hybrid with a planetary gear system by developing some blocks of his own and making some changes to ADVISOR's default layout [7].

D. *Availability*

An important goal in the development of ADVISOR was to make the entire simulator, including the source code, available to the public for free through the Web (www.nrel.gov/transportation/analysis). There are several reasons that DOE and NREL wanted to share ADVISOR with the public. The primary reason was to facilitate a sharing of component and model information amongst the advanced vehicle simulation community, reducing repeated duplication of effort. We had seen that many companies and organizations had a need for a tool such as ADVISOR, and that we would gain more feedback, validation, and new component models and data by first providing a common tool for other people to use without cost.

In the 9 months since we first made the tool available, we have seen an outpouring of interest from people who obtained ADVISOR and have been using it extensively. In fact, between September 1998 and July of 1999, nearly 800 people representing over 45 countries have downloaded ADVISOR. The list of people who have downloaded ADVISOR includes representatives from all of the world's major auto manufacturers, their major suppliers, universities, and small companies. Figure 11 shows the large geographic spread of the interest in ADVISOR, with each dot representing a person who has downloaded ADVISOR. The darkly shaded states and countries indicate that there is at least one person from that state or country has downloaded the software. Note the high concentration of people who have downloaded ADVISOR in the Detroit area, California, the Midwest, the East Coast, Europe, and Asia.

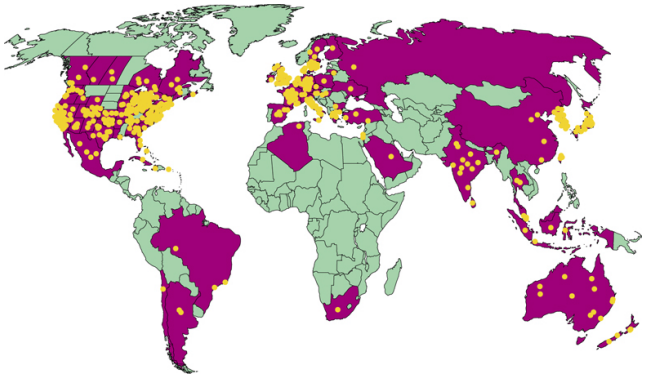

Figure 11. Geographic distribution of ADVISOR downloads as of June 1999

In addition to simply making ADVISOR available to the public, we also wanted to provide a place for people to ask questions about ADVISOR, and exchange ideas, data, and files, so our Web site has a Forum and a file-exchange area. We have found that many people are willing to freely share with NREL the data that they have entered into ADVISOR so that we may in turn share it with the rest of the current and future ADVISOR users. NREL is using the enthusiasm of the ADVISOR user community to improve the model through continually validating the simulator, expanding the database of component data and models, improving the

functionality and usability of the simulator, and ultimately feeding this back to the ADVISOR users.

E. Ease of use

Because we knew the potential users of ADVISOR 2.1 would have diverse backgrounds, we had to make the model easy to use so it would be accessible to such a large audience. Having a powerful and user-friendly GUI was the key to enabling ADVISOR to reach out to a broad audience and enable people to effectively answer vehicle systems analysis questions of their own. NREL wrote the GUI in the latest MATLAB®/Simulink® environment, and it enables straightforward access to many powerful analysis functions. The following descriptions of the three main GUI pages explain the wide array of features that are available for configuring a vehicle, conducting a simulation, and analyzing the results.

1) Vehicle Input Page

The layout of this screen is typical of all three GUI screens, in that the left-hand side of the window is the graphical representation of vehicle information and the right-hand side is where the user takes action. On the right-hand side of the screen, the user specifies what he wants to see and do to the vehicle, and controls the next action for ADVISOR to take. For example, on the vehicle input screen (see Figure 12), the picture in the upper left serves as a graphical indication of which vehicle configuration has been selected (conventional, series, parallel, fuel cell, or electric vehicle). The user-selectable graphs in the lower left allow the user to immediately view the performance information on the components that have been selected, such as efficiency contours for the engine and motor, emissions contours, and performance graphs for the batteries.

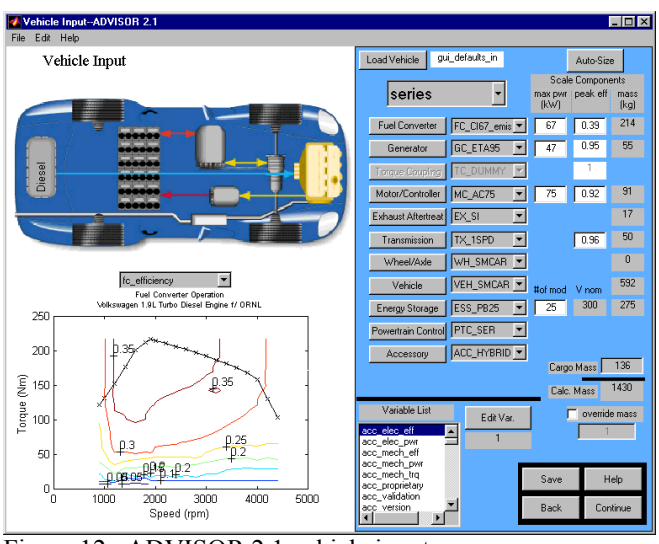

Figure 12. ADVISOR 2.1 vehicle input screen

On the right-hand portion of the vehicle input screen, the user controls what type of vehicle is simulated and the details of all the components that make up the drive system. Each component has a pull-down menu that allows different components to be selected from the ADVISOR library. The two columns of numbers under the "maximum power" and "peak efficiency" headings initially indicate these values from the data files, but typing in a new number causes the GUI to linearly rescale the entire efficiency map to match that peak efficiency while preserving the map's original shape. For example, entering in a 0.45 rather than the existing 0.42 in the engine peak efficiency would allow the user to examine the impact of a hypothetical engine that could achieve a 45% peak efficiency rather than 42%.

Just above these columns is an "autosize" button that simplifies the task of iteratively sizing drivetrain components (engine, motor, and batteries) to meet user defined minimum performance requirements of acceleration and gradability. For parallel vehicles, the autosize function also allows the user to select the degree of hybridization, which is reflected in the relative sizing of the engine, motor, and batteries.

Finally, the user can modify any scalar parameter that ADVISOR defines on the MATLAB® workspace through the variable list. Because the total vehicle test mass is a parameter that is often desirable to override in various "what-if" scenarios, it is brought to the top level and can be overridden with a single mouse click and by entering the new mass. Vehicle input files can also be saved; they store the names of the component files selected and all scaling and override settings to allow the user to recreate results at a future time or share input settings with a colleague.

2) Simulation Setup Page

The second of the three ADVISOR 2.1 GUI screens is the Simulation Setup screen (Figure 13). The primary decision for the user on this screen is whether to run a single cycle (and which one) or a test procedure, which can consist of special initial conditions, multiple cycles, and significant post-processing (such as the test procedure to determine combined city/highway fuel economy).

If the single cycle option is chosen, initial conditions (primarily thermal and battery) can be set, and for hybrids the type of battery state of charge (SOC) correction routine can also be selected. The two SOC correction options available are a zero-delta or a linear correction routine. The zero-delta routine iterates on the initial SOC until the final SOC is within some tolerance (0.5%) of it. The linear correction routine starts the battery at both its extreme high and low SOC, and then performs a linear interpolation to estimate the fuel economy at the zero-delta SOC crossing. Additionally, gradability and acceleration tests can be selected for evaluation.

Finally, because parametric studies are often useful to explore the design space of a given vehicle, ADVISOR 2.1 allows the option of doing a 1-, 2-, or 3-parameter design sweep of any scalar value on the workspace. This allows the sensitivity of a vehicle to its various input parameters to be evaluated, not only on fuel economy, but also on performance.

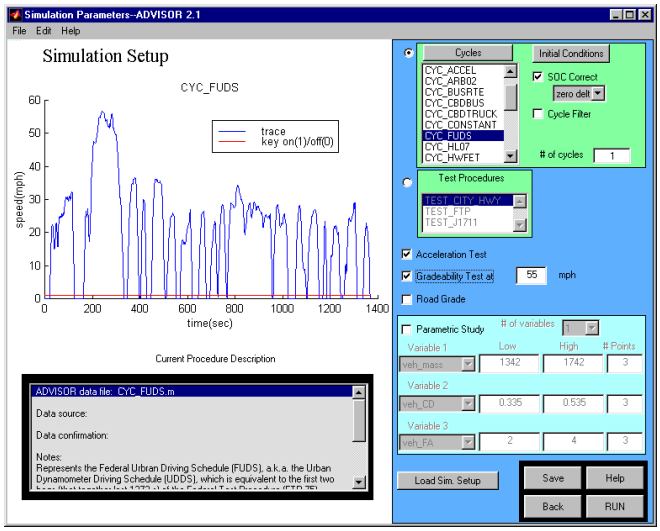

Figure 13. ADVISOR 2.1 simulation setup screen

3) *Results Page*

The Results Page (Figure 14) is the last of the three major ADVISOR screens. This page allows the user to see the summary results of fuel economy, emissions, acceleration, and gradability on the right-hand side, and plots of any of the time-dependent variables that the simulation puts onto the workspace on the left-hand side.

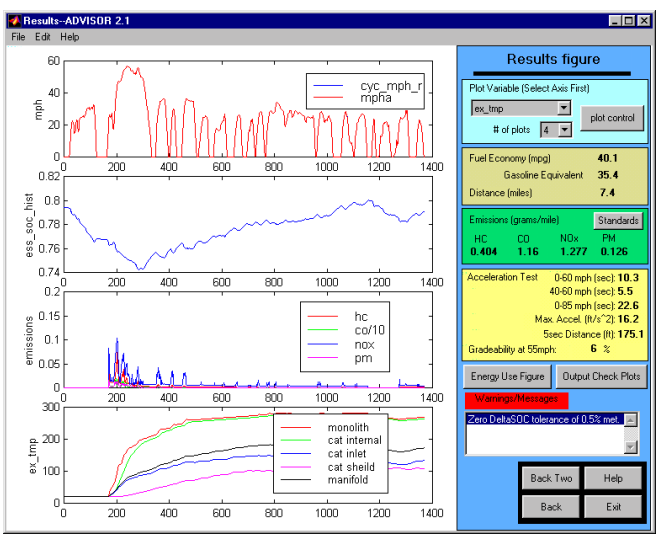

Figure 14. ADVISOR 2.1 Results Screen

The results screen has separate pop-up windows if test procedures or parametric studies are selected rather than single cycles. ADVISOR 2.1 allows full usage of the built-in plotting features of MATLAB® including zoom, layering multiple curves on the same graph, and applying gridlines. In Figure 14, which shows a sample results screen, you can see four representative plots: vehicle speed, battery SOC, regulated emissions, and temperatures at various places within the exhaust system.

There are two action buttons that pull up an energy usage figure and a series of diagnostic plots. The energy usage figure tracks all the energy through the drivetrain, notes where it is used, and performs an energy balance to make sure that there is no unaccounted-for energy. On all screens, there is a 'HELP' button that takes the user directly to the browser-viewable documentation for more information.

Feedback from our users indicates that we have been successful in creating a program that is easy to use and allows reasonably novice users to produce useful results as part of their vehicle systems analysis studies.

IV. Conclusions

The mathematical background behind ADVISOR 2.1, which uses a unique combined backward/forward facing approach, has been illustrated. The model and overall approach has been evaluated relative to the NREL team's objectives, which include accuracy, speed, flexibility, availability, and ease of use.

- Accuracy: For three vehicles ranging in 100 km/h acceleration time from 15 to 25 s, ADVISOR predicts to within 0.8% the acceleration time computed by iteration. Energy use on the US06 is predicted to within 1.9%, even for extremely underpowered vehicles that push the assumptions of ADVISOR's approach. For typical vehicles simulated, the energy predictions are within 0.02%.
- Speed: ADVISOR is 2.6 to 8.0 times faster than a comparable Simulink®-based forward-facing vehicle performance simulator in the tests documented here.
- Flexibility: The ADVISOR library contains numerous interchangeable component data files that may be used in a number of drivetrain configurations. It may be challenging to develop completely new drivetrain configuration models, but many ADVISOR users have already done this successfully.
- Availability: ADVISOR is available on the Web and has been downloaded by an international group of over 800 users representing over 45 countries.
- Ease of use: ADVISOR 2.1 includes a powerful GUI to allow even novice users to quickly analyze vehicle powertrains.

Symbols

F	force, N
I	current, A
J	rotational inertia, kg-m^2
m	mass, kg
P	power, W
r	radius, m
t	(simulation) time, s
v	vehicle speed, m/s
V	voltage, V
τ	torque, N-m
ω	rotational speed, s^{-1}

SUBSCRIPTS

act	actual
avail	available—possible given the drivetrain limits
con	associated with the motor controller
lim	subject to a component performance limit
map	computed from component performance map
mot	associated with the motor or motor/controller set
prev	computed in the previous time step
req	required
wh	associated with the wheel or wheel & axle

ACKNOWLEDGMENT

The authors gratefully acknowledge the continued support of the U.S. Department of Energy for vehicle systems analysis at NREL.

REFERENCES

[1] Cole, G. H., "SIMPLEV: A Simple Electric Vehicle Simulation Program, Version 2.0." EG&G Idaho, Inc. April 1993.

[2] "The AeroVironment Electric/Hybrid Vehicle Simulator, CarSim 2.5.4, Desktop Version, Documentation." AeroVironment, Inc. Monrovia, CA. August, 1994.

[3] Murrell, J. D., "Vehicle Powertrain Modeling," Letter Report under Consultant Agreement CCD-4-1403-01 to NREL, March 1995.

[4] Gillespie, T. D., Fundamentals of Vehicle Dynamics, published by Society of Automotive Engineers, Warrendale, PA, 1992.

[5] Bakker, E., H. B. Pacejka, L. Lidner, "A New Tire Model with an Application in Vehicle Dynamics Studies," SAE Technical Paper Number 890087, 1989.

[6] Senger, R. D., M. A. Merkle, D. J. Nelson, "Validation of ADVISOR as a Simulation Tool for a Series Hybrid Electric Vehicle," Technology for Electric and Hybrid Vehicles, Proceedings of the 1998 SAE Intl. Congress, Detroit, MI, Feb. 23-26, SAE Paper No. 981133, SP-1331, pp. 95-115, 1998.

[7] Santoro, M., "A Hybrid-Propulsion Powertrain with Planetary Gear Set for a 4WD Vehicle: Analysis of Power Flows and Energy Efficiency," 54th ATI (Associazione Termotecnica Italiana) National Congress, September 14-17, 1999, L'Aquila, Italy.

Keith B. Wipke holds a B.S. in mechanical engineering from the University of California, Santa Barbara, and an M.S.M.E. from Stanford University. He has worked at NREL since 1991, on activities such as testing and data collection from electric and hybrid vehicles and buses, computer modeling and optimization of hybrid vehicles using ADVISOR, and thermal and electric testing of hybrid electric vehicle batteries. He is currently the team leader for NREL's vehicle systems analysis team, which improves and maintains the ADVISOR advanced vehicle simulator.

Matthew R. Cuddy holds a B.S.M.E. from Cornell University and an M.S.M.E. from the University of Colorado. In 1992 and 1993, he developed models of turbulent diffuser and room airflow for buildings applications at NREL. From 1994 to 1998 he studied hybrid vehicle powertrain performance at NREL, developing ADVISOR in 1994. He is currently analyzing and developing new powertrain components and vehicle auxiliary load reduction systems as a consultant based in Amherst, Massachusetts.

Steven D. Burch holds an M.S.M.E. from Purdue University. Before joining NREL in 1992, he spent 4 years with Delphi - Harrison Thermal Systems (General Motors) as a project engineer in heat exchanger R&D. In his tenure at NREL, he developed numerous models of thermal systems in buildings and automobiles and also developed thermal management systems for batteries and catalytic converters. He was project leader for work with Benteler Automotive, Inc., to develop a practical vacuum-insulated automotive catalytic converter on which he holds a patent. He has recently joined General Motors' fuel cell division.