

ADVISOR: a systems analysis tool for advanced vehicle modeling

T. Markel^{*}, A. Brooker, T. Hendricks, V. Johnson, K. Kelly,
B. Kramer, M. O’Keefe, S. Sprik, K. Wipke

National Renewable Energy Laboratory, Golden, CO 80401, USA

Abstract

This paper provides an overview of Advanced Vehicle Simulator (ADVISOR)—the US Department of Energy’s (DOE’s) ADVISOR written in the MATLAB/Simulink environment and developed by the National Renewable Energy Laboratory. ADVISOR provides the vehicle engineering community with an easy-to-use, flexible, yet robust and supported analysis package for advanced vehicle modeling. It is primarily used to quantify the fuel economy, the performance, and the emissions of vehicles that use alternative technologies including fuel cells, batteries, electric motors, and internal combustion engines in hybrid (i.e. multiple power sources) configurations. It excels at quantifying the relative change that can be expected due to the implementation of technology compared to a baseline scenario. ADVISOR’s capabilities and limitations are presented and the power source models that are included in ADVISOR are discussed. Finally, several applications of the tool are presented to highlight ADVISOR’s functionality. The content of this paper is based on a presentation made at the ‘Development of Advanced Battery Engineering Models’ workshop held in Crystal City, Virginia in August 2001.

© 2002 Elsevier Science B.V. All rights reserved.

Keywords: ADVISOR; Systems analysis; Vehicle modeling; Vehicle simulation; Hybrid electric vehicle; Fuel cell; Battery; Energy storage

1. Introduction

Advanced Vehicle Simulator (ADVISOR) was first developed in November 1994 at the National Renewable Energy Laboratory. It was designed as an analysis tool to assist the US Department of Energy (DOE) in developing technologies for hybrid electric vehicles (HEV) through the Hybrid Electric Vehicle Propulsion System contracts with Ford, General Motors, and DaimlerChrysler. Its primary role is to highlight the system-level interactions of hybrid and electric vehicle components and their impacts on the vehicle performance and fuel economy.

ADVISOR was first released publicly via the Internet (www.nrel.gov/transportation/analysis) in September 1998. The number of public users of the software tool has grown steadily from a small group of about 30 initial partners to a group of over 4500 individuals that have downloaded the software. The majority (68%) of ADVISOR users are members of industry while 29% are members of academia with the remaining 3% representing government entities. This large user community provides feedback necessary to improve the models and component data to be used within the models. Since first introduced, several updated versions

of the software have been periodically distributed via the Internet. ADVISOR Version 3.2 is the latest version of the software.

This paper will discuss the fundamental approach used to model vehicle systems in ADVISOR. A description of ADVISOR’s capabilities and limitations will be presented. ADVISOR has three types of power source models that can be included in a vehicle model. These include an internal combustion engine, a fuel cell system, and an energy storage system. The general modeling approach employed for each of these systems will be described briefly. The paper will also highlight typical applications of the tools to real systems analysis problems.

2. ADVISOR structure

ADVISOR was created in the MATLAB/Simulink environment. MATLAB provides an easy-to-use matrix-based programming environment for performing calculations while Simulink can be used to represent complex systems graphically using block diagrams.

ADVISOR uses three primary graphical user interface (GUI) screens to guide the user through the simulation process. With the GUIs, the user can iteratively evaluate the impacts of vehicle parameters and drive cycle requirements

^{*} Corresponding author. Tel.: +1-303-275-4478; fax: +1-303-275-4415.
E-mail address: tony_markel@nrel.gov (T. Markel).

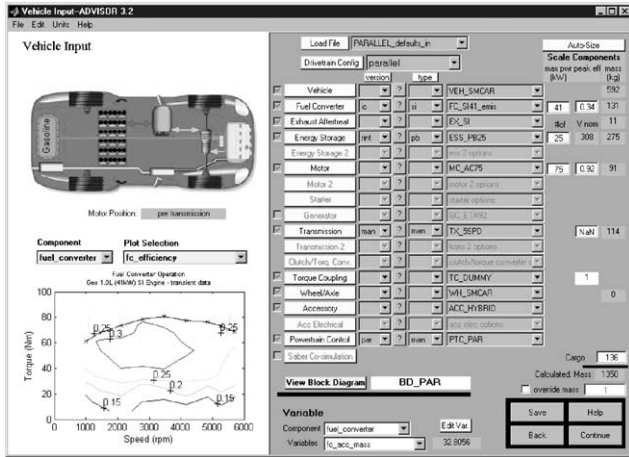


Fig. 1. ADVISOR vehicle input window.

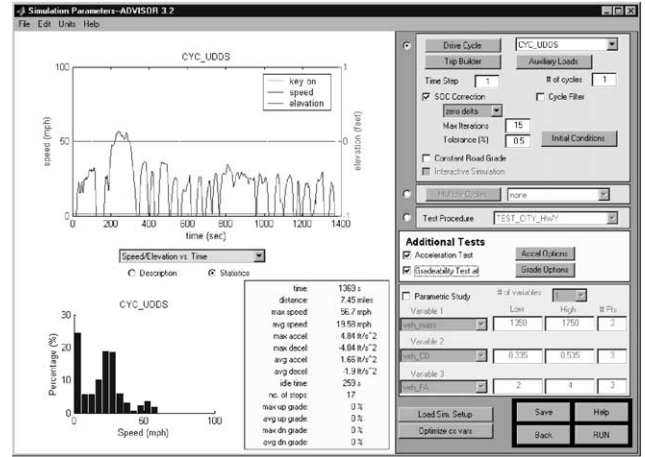


Fig. 2. ADVISOR simulation setup window.

on the vehicle performance, fuel economy, and emissions. The GUIs facilitate interaction with the raw input and output data that is present in the MATLAB workspace. The vehicle model is depicted graphically using Simulink block diagrams to define the connections between components. The model then reads the input data from the MATLAB workspace during the simulation and outputs the results to the workspace to be viewed in the results window.

In the ADVISOR vehicle input window, shown in Fig. 1, the user “builds” the vehicle of interest. Pull-down menus are used to select a vehicle configuration (i.e. series, parallel, conventional, etc.), and the components that will compose the driveline. Characteristic performance maps for the various components are displayed in the lower left of the window and are accessible using the associated pull-down menus. The size of a component (e.g. peak power and number of modules) can be modified by editing the characteristic value displayed in the boxes on the far right portion of the screen. Lastly, any scalar parameter can be modified using the edit variable menu in the lower right portion of the window. All vehicle configuration parameters can be saved for future use. Once the user is satisfied with the vehicle input characteristics, the ‘continue’ button takes them to the simulation setup window.

In the ADVISOR simulation setup window (Fig. 2), the user defines the event over which the vehicle is to be simulated. Some of the events that may be simulated include a single drive cycle, multiple cycles, and special test procedures. Again, in the right portion of the window, the user selects cycles and defines the simulation parameters while in the left portion; information about the selections is provided. For example, when a single drive cycle is selected, the user can view the speed trace in the upper left portion and a statistical analysis of the cycle in the lower left portion. With the simulation parameters configured, clicking on ‘run’ will execute the simulation and provide a results screen at completion.

The ADVISOR results window (Fig. 3), provides the ability to review the vehicle performance, both integrated

over a cycle and instantaneously at any point in the cycle. On the right portion of the window, summary results such as fuel economy and emissions are provided. In the left portion, the detailed time-dependent results are plotted. The results displayed on the left can be dynamically changed to show other details (e.g. engine speed, engine torque, battery voltage, etc.) using the pull-down menus in the upper right portion of the window.

The ADVISOR GUI is used to interact with the data in the MATLAB workspace. Component data is stored in the form of text files as shown in Fig. 4. Based on the user’s selections, the appropriate data sets are loaded into the workspace. The GUI is also used to control the selection of the model to be used. The model, displayed as a graphical block diagram in Figs. 4 and 5, reads the data that has been loaded into the MATLAB workspace as an input parameter set.

Finally, the actual vehicle model is composed of a collection of component models. In Fig. 5, it can be seen that the individual component models are stored in a library. The component models can be inserted into a vehicle model and then connected to define the flow of torque/speed and power

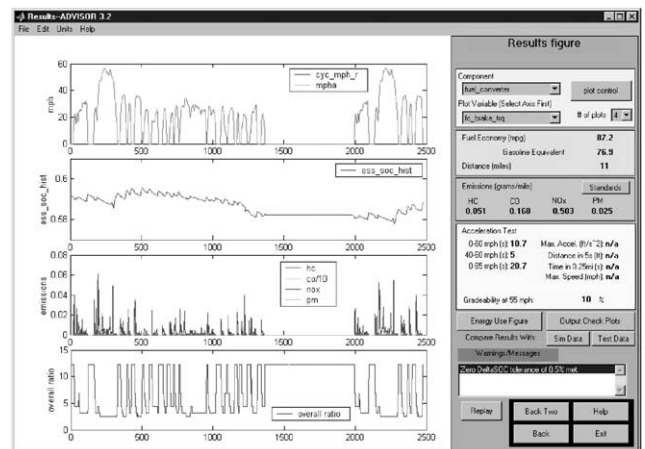


Fig. 3. ADVISOR results window.

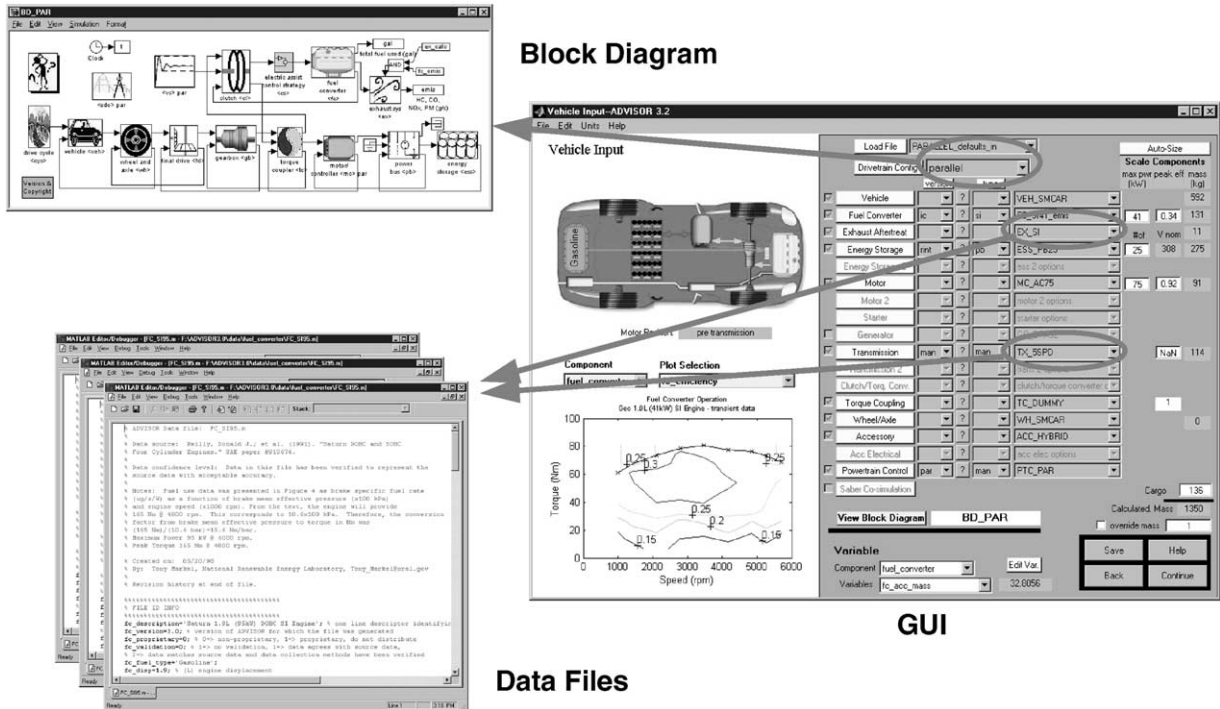


Fig. 4. ADVISOR interaction between data files, GUI, and model (block diagram).

from one component to the next. The model library approach allows the same component model to be reused in multiple vehicle configurations. It also allows the impacts of different models (e.g. simple versus detailed) to be evaluated within a single vehicle architecture.

The arrows entering the top input of a component block in the vehicle model in Fig. 5 depict a torque and speed or a

power request from one component to the next upstream component. The request is initially based on the vehicle speed requirements and is modified as it passes through each component to account for losses associated with each component. Arrows entering the bottom input port of each block represent what the upstream component is able to achieve. Typically, this will be the same as that requested unless a

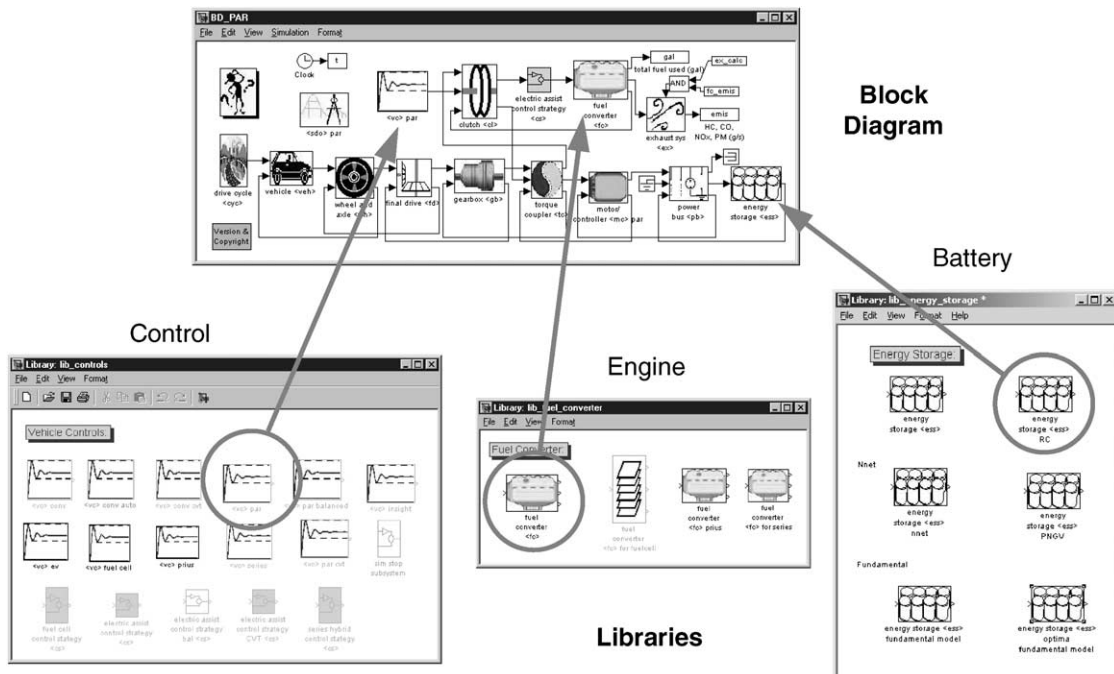


Fig. 5. ADVISOR model definition employs linked library architecture.

component limit or control feature has been encountered. Component performance limitations (e.g. minimum battery operating voltage) are accounted for within the individual component controls and may impact the achievable performance.

3. Approach

ADVISOR employs a unique combination of backward- and forward-facing simulation attributes. A purely backward-facing simulation propagates a high level requirement (e.g. change from X to Y speed in Z seconds) linearly backward through a series of systems (e.g. vehicle \Rightarrow wheels \Rightarrow transmission \Rightarrow engine). While a forward-facing approach iteratively modifies individual component control commands to the various vehicle subsystems in an effort to find the combination that minimizes the error between the driver demand and the actual response of the system to the control commands. Each has advantages and disadvantages and each excels in their intended applications. A detailed discussion of the backward/forward-facing approach is presented in [1] through an example calculation.

3.1. 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 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 backward-facing approach is convenient because automotive drivetrain components tend to be tested in a laboratory environment such that a table of efficiency or loss versus output torque and speed (or power) 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 (e.g. 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. Since the backward-facing approach assumes that the trace is met, this approach is not well suited to compute 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 directly measurable in a vehicle. For example, control signals such as throttle and brake position are absent from the model, further hindering dynamic system simulation and detailed control system development.

3.2. 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 $a = F/m_{\text{eff}}$, 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. Since forward-facing models deal in quantities measurable in a actual 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. As a result, forward-facing simulation can be overly time-consuming for use in preliminary design studies.

3.3. Combined backward/forward-facing approach

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- and forward-facing approaches are as follows:

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.

More information on the backward/forward-facing approach as implemented in ADVISOR can be found in [1].

4. Capabilities and limitations

ADVISOR was originally developed as a simple analysis tool that could be used to quickly quantify the relative impacts of advanced technologies in a vehicle. It quickly evolved into a tool with a wide range of capabilities.

4.1. Key attributes

The following is a short list of the key attributes that have lead to the adoption of ADVISOR as an analysis tool by a broad audience:

- Intuitive, easy-to-use graphical interface;
- Fast solutions;
- Distributed as open source code;
- Customizable;
- Scalable component models;
- Good customer support, software maintenance, and documentation;
- Free and publicly available;
- Highly parameterized models;
- Provides robust solutions;
- Modular architecture.

Prior to developing ADVISOR, other simulation tools were considered [2–4]. To support the DOE efforts, ADVISOR was designed to be flexible and open such that new technologies, unique energy management strategies, and alternative vehicle configurations could be easily incorporated into and evaluated within a system architecture. The user receives all of the source code when the package is downloaded.

The open architecture and availability of source code allows a significant amount of customization. Users can replace the existing component models with more detailed models if necessary. Simulink makes it possible to link to other software packages for component models. Proprietary models can be compiled and linked to Simulink to protect intellectual property.

The ADVISOR GUI is laid out in a very intuitive manner and provides the ability to easily and quickly vary parameters and evaluate many different vehicle scenarios. Likewise, the robustness and repeatability of the solutions provided by ADVISOR greatly enhances its reputation as an unofficial “industry standard”.

ADVISOR was first distributed as a free and publicly available tool via the Internet in September 1998. Customer support for user questions and maintenance of the software have been important attributes that have helped to gain and maintain the trust of those using the software for industry vehicle systems analysis projects.

Finally, nearly everything in ADVISOR has been parameterized. As a result, components can be scaled easily to produce new vehicles that can be compared to baseline scenarios. Optimization routines have been wrapped around parameterized models to highlight opportunities for improved vehicle design.

4.2. Functionality

The two most common simulations performed for a vehicle in ADVISOR include drive cycle analysis and performance tests. A drive cycle constitutes a series of vehicle speeds as a function of time. There are more than 40 different drive cycles to choose from in the ADVISOR database. Some of the drive cycles even have roadway grade associated with them like the NREL to Vail, Colorado driving cycle (CYC_NREL2VAIL) provided below in Fig. 6. This data was collected by NREL engineers using on-board data acquisition equipment and can be used with the vehicle model to verify a vehicle’s operating characteristics in a “real world” driving scenario of crossing the Continental Divide by interstate highway.

A performance test allows the user to assess the acceleration and gradeability performance of a vehicle. The test routines provide many customizable parameter settings such as running the test with or without the battery pack enabled for hybrid vehicles. They can also be run at test weights other than the actual vehicle weight. The performance tests have been formatted to provide a significant amount of flexibility in determining how the vehicle performance will be assessed. Fig. 7 shows how a typical vehicle would

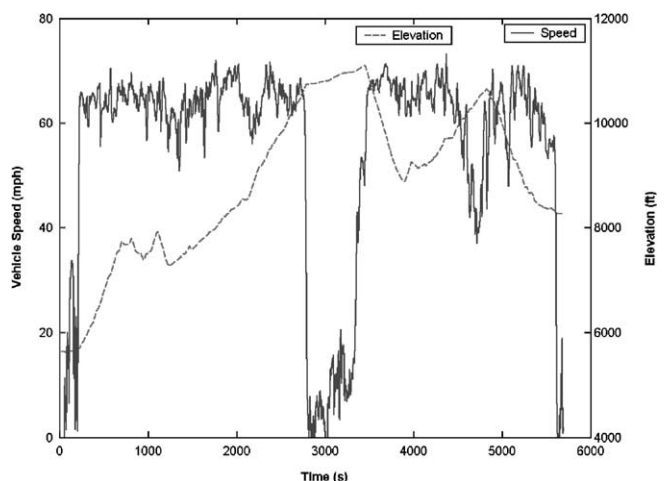


Fig. 6. NREL to Vail, Colorado drive cycle.

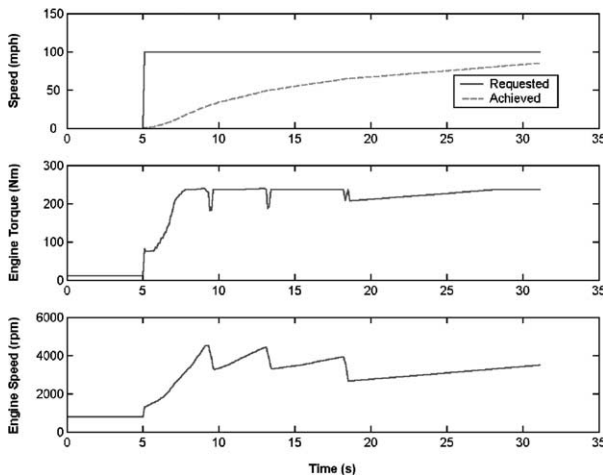


Fig. 7. Acceleration test results for a typical conventional vehicle.

respond during a maximum effort acceleration event in ADVISOR.

Special test procedures have also been developed that build upon the standard drive cycle analysis by analyzing multiple cycles at one time and/or performing special calculations based on the results. For example, the City/Highway Test (TEST_CITY_HWY) (1) initializes the system to hot ambient conditions, (2) runs a highway cycle, (3) stores the results, (4) initializes the system to cold ambient conditions, (5) runs an FTP, (6) stores the results, and (7) calculates the combined City/Highway fuel economy value. Other available test procedures include the following:

- Grade Test Procedure;
- Acceleration Test Procedure;
- FTP Test Procedure;
- FTP Test Procedure for hybrids;
- City/Highway Test Procedure;
- City/Highway Test Procedure for hybrids;
- SAE J1711 Test Procedure;
- “Real World” Test Procedure.

State of charge (SOC) balancing is an important aspect of hybrid vehicle analysis. If the change in SOC of the battery between the beginning and the end of a cycle is too large, the vehicle fuel economy may be artificially very high or very low due to the battery net discharge or charge, respectively. ADVISOR offers two methods for ensuring SOC-balanced vehicle results over a drive cycle such that multiple simulation scenarios can be compared on a consistent basis.

The first method uses a linear approximation approach. The vehicle is simulated first when starting from a high SOC then from a low SOC. Linear interpolation of fuel economy and emissions between the two simulations is used to determine the results at zero change in SOC.

The second method is an iterative zero-delta approach. ADVISOR iteratively changes the initial conditions of the battery pack until the final state is within a specified tolerance of the initial state. Although, the zero-delta approach will

typically require more simulations than the linear method, it ensures that the results are “real” and physically possible rather than mathematical estimates as they are with the linear approximations method.

In future versions, ADVISOR will also include a SOC balancing routine that will be based on ensuring that the equivalent fuel energy of the change in SOC of the battery pack is less than a specified percentage of the total fuel consumed during a cycle. This approach is documented in SAE J1711 [5] and will eliminate fluctuations in results due to variations in total battery pack capacity among vehicles (e.g. 5% SOC change in a 50 Ah pack is significantly more energy than a 5% SOC change in a 5 Ah battery pack).

4.3. Limitations

ADVISOR was developed as an analysis tool, and not a design tool. Its component models are quasi-static, and should not be used alone to predict phenomena with very small time scales. For example, ADVISOR should not be used to quantify physical vibrations, electric field oscillations and other fast dynamics. ADVISOR, however, has been successfully linked with other tools that do perform dynamic analysis capability. Tools like Saber for electrical systems modeling and ADAMS/Car for vehicle dynamics have been linked with ADVISOR. These dynamic modeling tools focus on only a portion of the analysis while ADVISOR simulates the rest of the vehicle. For example, a battery modeled in Saber can be configured to communicate periodically with the rest of vehicle systems in ADVISOR during a drive cycle simulation.

As an analysis tool, ADVISOR uses the required vehicle speed as an input to determine what drivetrain torques, speeds, and powers would be required to meet that vehicle speed. Because of this flow of information back through the drivetrain, from tire to axle to gearbox and so on, ADVISOR can be classified as a backward-facing vehicle simulation with the added ability to evaluate wide-open throttle operation without the need for iteration.

Forward-facing vehicle simulations include a model of a driver that monitors the required speed and responds with an accelerator or brake position, to which the drivetrain responds with a torque. This type of simulation is well suited to the design of control systems, for example, down to the integrated circuit and PC card level—the implementation level.

ADVISOR is well suited to evaluate and design control logic by iterative evaluation. Control logic includes things like “when the engine torque output is low and the battery SOC is high, turn off the engine.” The control logic, with which ADVISOR can work, is about what you want the vehicle to do. The control system, beyond ADVISOR’s purview, relates to the details of how to make the vehicle and each component do what it needs to do in order to meet the demands of the control logic.

5. Power source models

The ADVISOR component models, in general, are empirical models based on test data. Data tables developed from experimental data or other models are used to store efficiency, fuel consumption, emissions, and/or losses indexed by the operating characteristics of the system (e.g. speed and torque for an engine). The input data is typically based on steady-state component testing. Transient operation is estimated by a series of small steady-state steps. The efficiency or the loss associated with a component is then determined by interpolating between known operating points. ADVISOR has three power source models that can be included in vehicle models. They represent an internal combustion engine, a fuel cell system, and a battery pack. Each of these will be discussed briefly in the following sections.

5.1. Internal combustion engine

The internal combustion engine model in ADVISOR accepts an input command for the torque and speed requested by the driveline or vehicle controller. Via interpolation of the input fuel consumption data and the engine-out emissions data at the specified speed and load, the resulting fuel consumption and engine-out emissions can be calculated for the current time step. Losses due to changes in rotational inertia and mechanical accessories are accounted for within the component model. Fig. 8 provides an example of a typical fuel consumption input data set.

Interpolation of fuel use and emissions data is performed on hot steady-state data. Temperature correction factors are included to account for operation prior to the engine reaching hot steady-state conditions. The correction factors can either be map based (vary with speed, load, and temperature) or they can be equation based (vary with temperature only).

Since the fuel consumption and the engine-out emissions will be adjusted based on engine temperatures, it is important that accurate tracking of engine temperatures is maintained.

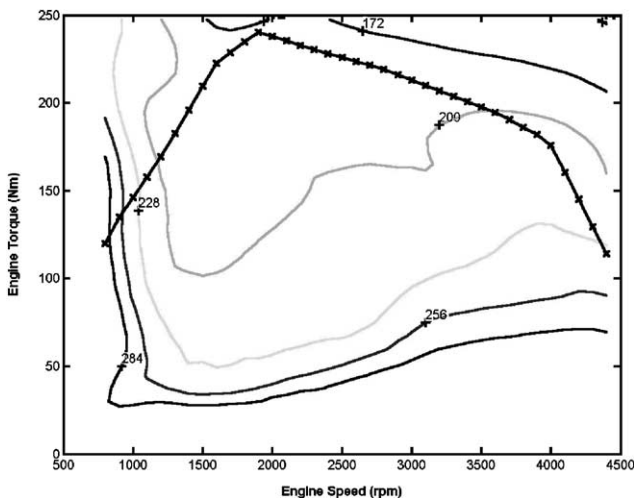


Fig. 8. Typical engine fuel consumption data.

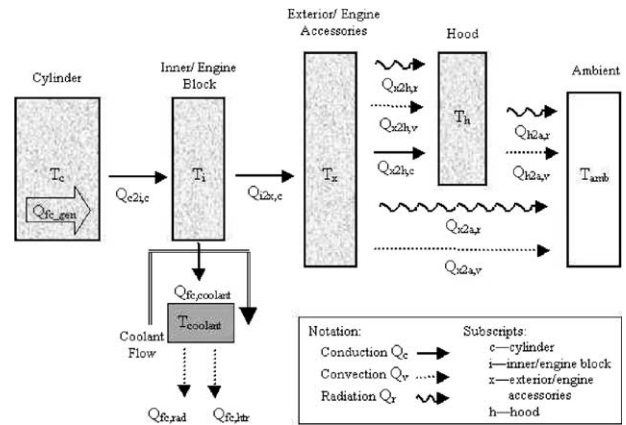


Fig. 9. Engine thermal network model.

The thermal model of the engine assumes that the engine is a multi-node lumped-capacitance thermal network. The total heat load through the system is equal to the difference between the shaft work output of the engine and the energy content of the fuel input to the system. A fraction of the waste heat is carried away with the exhaust stream while the remainder is dissipated among the various engine components and coolant flow. Fig. 9 depicts the nodes of the thermal network model and the heat transfer mechanisms considered. The engine-out emissions information is passed from the engine model to a catalyst model that calculates the heat transfer among the various nodes of the catalyst and the conversion efficiency for each constituent at the current catalyst temperature. The conversion efficiency terms for each emission constituent are used to determine the tailpipe-out emissions results.

5.2. Fuel cell power system

ADVISOR includes two empirically based fuel cell system modeling options and one co-simulation option. The first model simply lumps the entire fuel cell system into a single element with characteristic system efficiency as a function of net power out of the system. The system could be a simple ambient pressure hydrogen stack or it could be a complex gasoline reformed fuel cell system. The key assumption is that the system can provide a specific net power while consuming a set amount of fuel regardless of how complex the system maybe. A typical input data set for such a system is shown in Fig. 10.

The second modeling approach is similar except that the auxiliary systems (i.e. air compressor, fuel pump, cooling fans, etc.) performance can be specified separately from the fuel cell stack. The fuel cell stack performance is characterized with a polarization curve, the associated fuel usage per cell, and the number of individual cells within the stack. The balance-of-plant systems are specified separately and specify the electrical power required to provide the necessary flowrate to the fuel cell stack. In this case, the fuel cell stack must provide enough power to supply not only the driveline

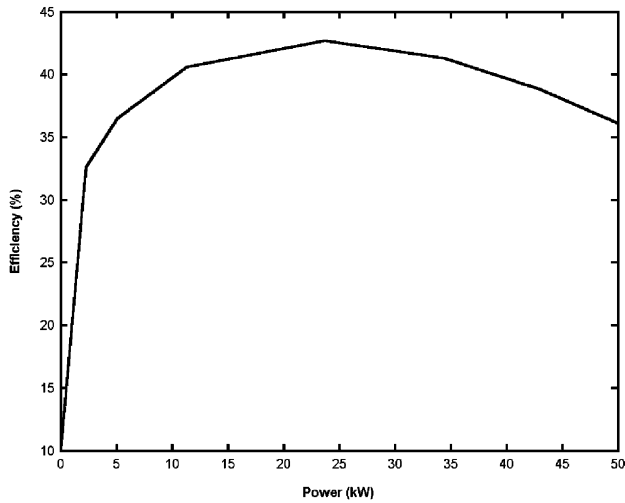


Fig. 10. Typical input data for net power vs. efficiency fuel cell system model.

needs but also enough to power its own accessory systems. The net power remaining after accessory loads have been accounted for can be provided to the driveline. Figs. 11 and 12 provide an example data set that may be used in this model.

Finally, the third fuel cell system model is achieved via a simulation link between ADVISOR and the General Computational Toolkit (GCTool). GCTool is a software package developed and maintained by Argonne National Laboratory for modeling complex thermodynamic and chemically reacting systems [6]. GCTool can represent systems from simple stacks all the way to complete systems including a reformer. When using GCTool with ADVISOR, the GCTool model is first called independently in a design mode to determine the system characteristics (i.e. fuel cell dimensions, flowrates, and pressures) necessary for the desired peak operating conditions. The configuration details are saved. During the simulation, these configuration parameters are held constant while the operating power level is varied

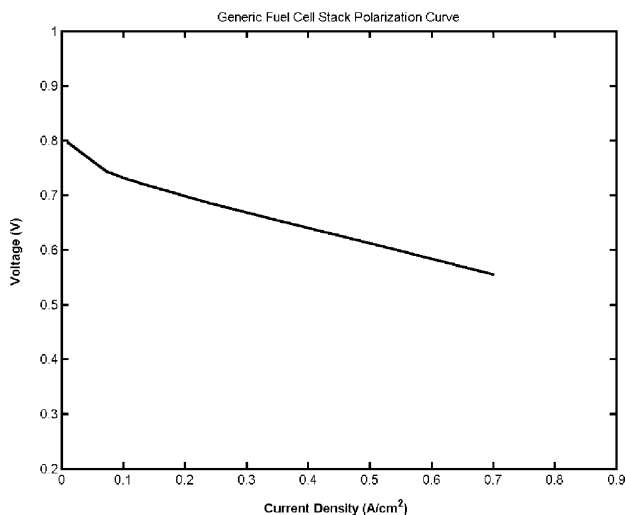


Fig. 11. Fuel cell stack polarization curve input data.

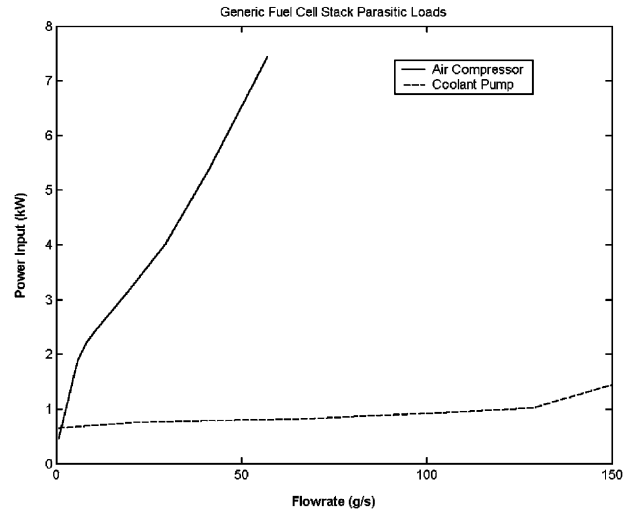


Fig. 12. Parasitic load input data for polarization curve model.

and the resulting power output and fuel usage are provided to ADVISOR.

The current fuel cell system modeling options in ADVISOR provide significant capabilities to evaluate fuel cell systems in hybrid and electric vehicles. However, we intend to improve accuracy and flexibility in the fuel cell modeling area in the future. The current development focus area is on the improvement of the thermal network model. The existing model makes use of the existing thermal network for an internal combustion engine with different parameter settings. A fuel cell system specific thermal network model will be available in the near future. Other fuel cell system models, including those with fuel reformer, compressor, and expander sub-model options are also under consideration for inclusion in future versions of ADVISOR.

The incorporation of three fuel cell system models of varying degrees of complexity discussed above provide a good example of ADVISOR’s ability to connect with models of various levels of detail. When it is of value to include additional detail, the user can choose to do so.

5.3. Energy storage system

Like the fuel cell power system, multiple energy storage system models have been developed and incorporated into ADVISOR. Brief explanations of the model options are provided later (for detailed discussions see [7]).

The first model whose equivalent circuit diagram is shown in Fig. 13 is based on a simple resistive circuit with a voltage source. This model is easy to build within the Simulink environment. It runs quite quickly and provides reasonable results. However, without a capacitor in the equivalent circuit, the load voltage can swing dramatically and result in premature enforcement of performance limits.

The second and third models, RC and PNGV, attempt to more closely represent the battery characteristics by including capacitors within the branches of the circuit as shown in

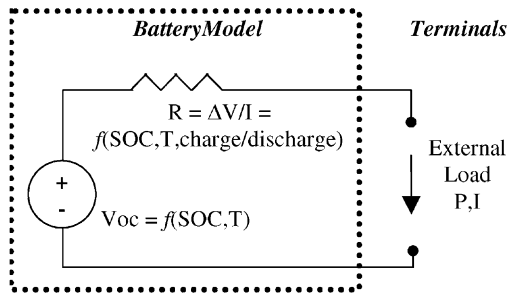


Fig. 13. Internal resistance battery model electrical schematic.

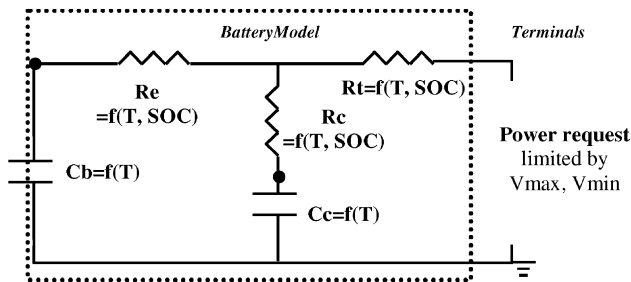


Fig. 14. Resistance–capacitance (RC) battery model electrical schematic.

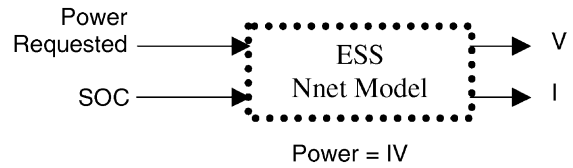


Fig. 16. Neural network battery model block diagram.

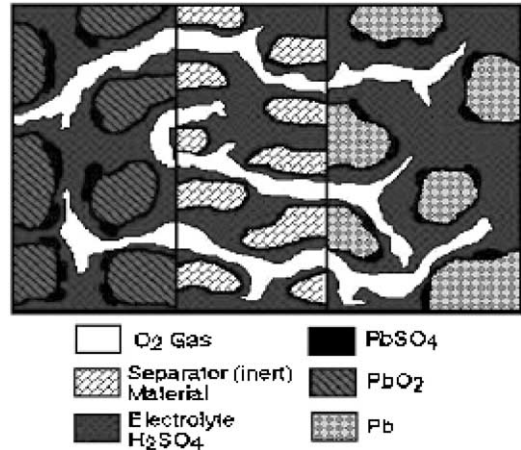


Fig. 17. Fundamental lead acid diagram.

As a result, the load voltage responds more smoothly and more like the real battery when a load is applied.

The fourth model is based on a neural network approach (Fig. 16). The application of neural networks to battery modeling allows the user to develop a model without having to perform extensive standardized battery testing. It allows the user to create a model with a limited data set. The disadvantage to this approach is that the model is only valid within the range for which test data has been collected and a significant amount of data is needed to ensure accurate predictions.

The fifth model type is a fundamental battery model (Fig. 17). As with GCTool for fuel cells, Simulink has been used in this case to link to a compiled and executable battery model. Prof. John Harb of Brigham Young University

developed the fundamental battery model [8]. Extensive knowledge of battery parameters is necessary to effectively use this model. However, it demonstrates the ease with which compiled and proprietary models can be linked into ADVISOR when more detail or customization is desired.

The majority of the input data for the battery models included with ADVISOR has been collected in the Battery Thermal Management Laboratory. This facility (Fig. 18) not only provides model input data but also the ability to validate the battery model results.

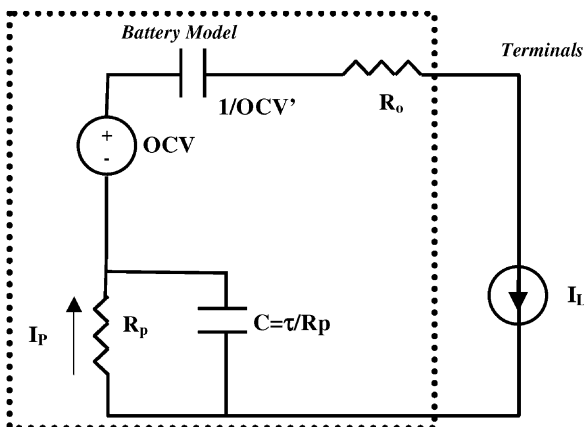


Fig. 15. PNGV battery model electrical schematic.



Fig. 18. NREL's Battery Thermal Management Laboratory.

The first three models discussed above use a standard lumped parameter thermal model. This model treats the entire pack of modules as a single mass with typical heat transfer and air flow properties that closely represent the system performance.

In the future, we expect that ADVISOR will also include links to Saber or other electrical systems modeling tools for modeling batteries and other electrical components. The MATLAB Power Systems Blockset has also be used to model batteries with a representative equivalent circuit without having to solve the circuit equations.

6. Applications

ADVISOR has been applied to a wide variety of vehicle system analysis problems. Typically, ADVISOR is used to help individuals and companies assess the impacts of their work in the vehicle systems environment. For example, if a battery developer was able to extend the voltage limitations of a battery, ADVISOR could be used to quantify how this extended performance range may provide additional power capability during operation. As a result, other components may be downsized saving both mass and cost, leading to improved vehicle fuel economy over a drive cycle.

ADVISOR has been used to develop and analyze technical targets for the DOE programs. The technical targets are used to guide future government research programs towards the most promising technology areas. ADVISOR can be used to derive the performance requirements for vehicle subsystems (e.g. motor peak power required during a cycle or the power profile that a battery will be expected to follow). The technical targets are development goals at the component level (i.e. improve the peak engine efficiency from 40 to 45%). Annually, a study is completed to confirm that if the output of all of the technology development programs were to be combined into a real vehicle, that the vehicle-level fuel economy and emissions goals could be achieved.

Vehicle system optimization is also an area in which ADVISOR excels. Various optimization algorithms have been wrapped around ADVISOR (Fig. 19) to automate the iterative process of design improvement. The solution

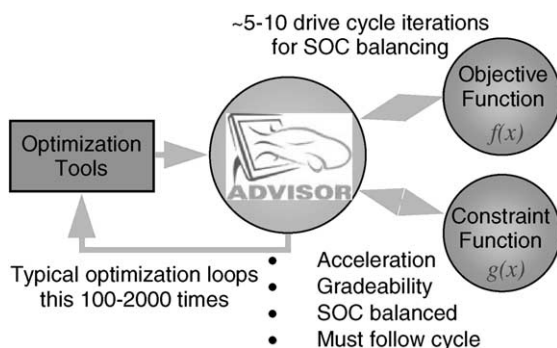


Fig. 19. ADVISOR linkage to optimization tools.

time for a single analysis is relatively short, on the order of 1/10 real time. As a result, many possible scenarios can be evaluated quickly. Recently, ADVISOR was used to determine the optimal component sizes and control strategy parameters for a fuel cell hybrid sport utility vehicle. Optimal sets of vehicle parameters were derived for a family of vehicles. Each of these vehicles was optimized with respect to fuel economy over the drive cycle of interest. It was found that the drive cycle, for which a vehicle is designed, has a significant influence on the optimal combination of component sizes and control strategy parameters. For this study, more than 35,000 parameter combinations were evaluated automatically over a period of approximately 1 week.

A component supplier or automobile manufacturer can use ADVISOR to quantify the system requirements of a specific component. Batteries are a key component for hybrid vehicles. They also represent a significant portion of the incremental cost. It is important to determine the minimum battery pack size or capacity that can meet the vehicle needs.

In a recent study, the duration over which the battery could sustain the vehicle electrical accessory loads while the engine was shut-down was quantified for a variety of battery sizes and accessory loading levels. From Fig. 20, it is shown that a 250 Wh battery pack could sustain a 1000 W load for approximately 10 min. However, if the load was 3000 W (typical of today’s air conditioning systems), the battery could only sustain the load for about 4 min. ADVISOR can be used to assess the duration of engine-off periods and track the usage of the battery for supplying accessory loads. With this information, automobile manufacturers and component suppliers can quickly determine capacity requirements and expected operating conditions.

The widespread application of hybrid vehicle technology is likely to be one of the key factors to improved vehicle fuel economy. ADVISOR has been used to help quantify the reduction in vehicle losses from both the implementation of hybrid technology and the application of various other

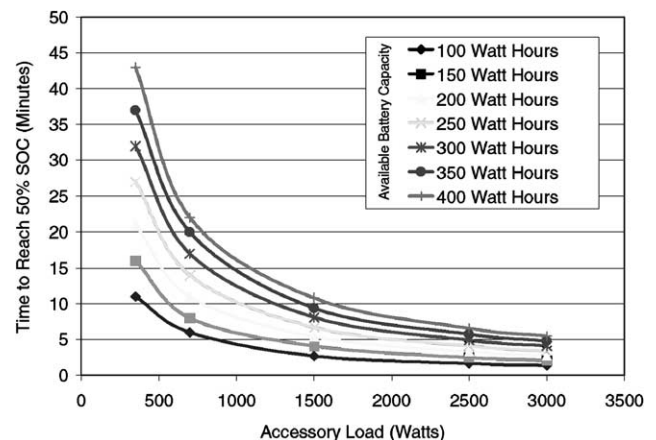


Fig. 20. Battery capacity required to support electrical accessory loads.

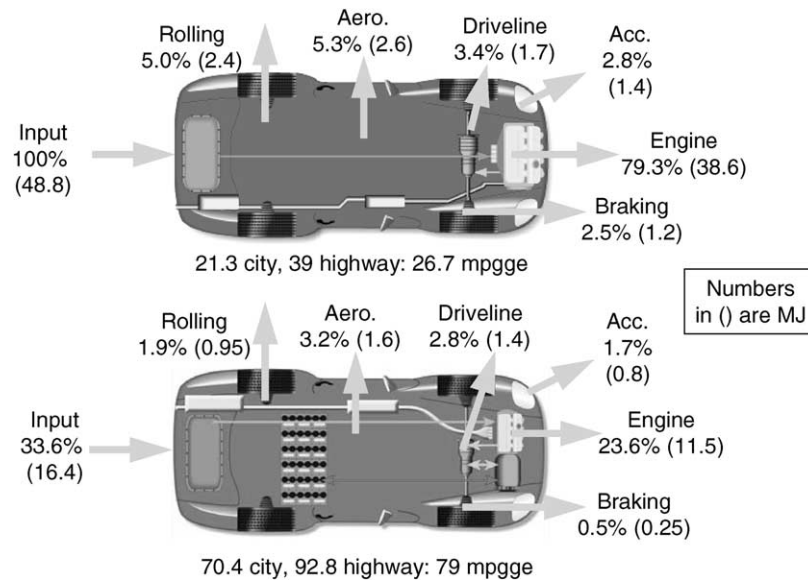


Fig. 21. Energy usage analysis of conventional and advanced technology vehicle.

technologies. Fig. 21 highlights the contributions of various advanced technologies to improved vehicle fuel economy and reduced system losses. First, a baseline conventional automatic transmission vehicle was derived that provided a combined¹ fuel economy of approximately 27 mpgge (mpg gasoline equivalent). An energy usage analysis is performed on the whole vehicle over the drive cycle. The results of this analysis for the conventional vehicle are shown in the top portion of Fig. 21. It is apparent that a large portion of the total energy input is consumed in the engine. Engines are typically inefficient during idling and low load conditions. Converting this vehicle to a parallel hybrid can have significant impact on these losses. In the lower portion of Fig. 21, the same energy usage analysis has been performed on a parallel hybrid vehicle that incorporates several technology improvements. Table 1 summarizes the assumed improvements over the baseline conventional case.

As a result of the applications of all of these technology improvements, the total energy input has been reduced by 66%. Many of the technology improvements simply reduce the load seen by the engine, while hybridization helps to eliminate engine losses due to idling and low load operation. Since the load has been reduced significantly and there is a battery pack and electric motor to supplement its capabilities, the engine size can be reduced significantly. Engine downsizing typically leads to increased overall operating efficiency.

Table 1

Technology improvements for energy balance analysis

Technology area	Improvement
Accessory loads	Reduced by 40%
Rolling resistance	Reduced by 40%
Engine efficiency	Increased by 30% (gasoline to diesel)
Driveline	Parallel hybrid electric driveline components (regenerative braking)
Vehicle mass	Reduced by 40%

An energy usage analysis can help focus technology development on areas that can have the greatest impact on energy conservation efforts. ADVISOR provides the capability to do this type of analysis quickly for advanced technology vehicles operating over entire drive cycles.

7. Conclusions

ADVISOR has been developed by the National Renewable Energy Laboratory for the US DOE. It is a tool that can be used to evaluate and quantify the vehicle level impacts of advanced technologies applied to vehicles. It is written in the MATLAB/Simulink environment and is freely distributed via the Internet. ADVISOR provides the vehicle engineering community with an easy-to-use and flexible, yet robust and supported analysis package for advanced vehicle modeling.

ADVISOR is primarily used to quantify the fuel economy, performance, and emissions of vehicles that use alternative technologies, specifically HEV architectures. It employs a unique combined backward/forward-facing modeling approach. This approach allows ADVISOR to accurately represent vehicle operation under a multitude of operating

¹ Combined fuel economy is calculated as

$$\text{combined FE} = \frac{1}{(0.55/\text{cold-FTP}) + (0.45/\text{hot-HWY})}$$

where cold-FTP represents cold-start city driving fuel economy and hot-HWY represents the hot-start highway driving fuel economy.

scenarios without the need to iterate, as other models must. ADVISOR's solution speed (on the order of 1/75th real time) makes it an excellent choice for vehicle system optimization studies.

When detailed component models are necessary, the open and modular design of ADVISOR makes the connection to detailed, dynamic, and proprietary models possible. ADVISOR currently includes multiple fuel cell models and battery models of varying degrees of complexity.

ADVISOR has been applied by researchers at NREL, industry, government, and academia to understand the impacts of various technologies on the performance, fuel economy and emissions of a vehicle. Typical applications include requirements definition, system optimization, and energy usage assessments.

Acknowledgements

The authors would like to acknowledge the US Department of Energy, specifically Robert Kost and Patrick Sutton for their support of ADVISOR. ADVISOR would not be what it is today without the large number of users that provide data, models, and feedback. Thanks to all of you. Lastly, we would like to recognize the former team members

that were instrumental in the early stages of development including Matt Cuddy (currently studying energy policy at Princeton) and Steve Burch (now with General Motors Advanced Powertrain Center).

References

- [1] K.B. Wipke, M.R. Cuddy, S.D. Burch, ADVISOR 2.1: a user-friendly advanced powertrain simulation using a combined backward/forward approach, *IEEE Transaction on Vehicular Technology* 48 (6) (1999) 1751–1761.
- [2] G.H. Cole, *SIMPLEV: A Simple Electric Vehicle Simulation Program*, Version 2.0, EG&G Idaho Inc., April 1993.
- [3] *The AeroVironment Electric/Hybrid Vehicle Simulator, CarSim 2.5.4, Desktop Version, Documentation*, AeroVironment Inc., Monrovia, CA, August 1994.
- [4] J.D. Murrell, *Vehicle Powertrain Modeling*, Letter Report under Consultant Agreement CCD-4-1403-01 to NREL, March 1995.
- [5] *Recommended Practice for Measuring the Exhaust Emissions and Fuel Economy of Hybrid Electric Vehicles*, SAE J1711 Standard, March 1999.
- [6] *General Computational Toolkit (GCtool) Documentation*, Argonne National Laboratory.
- [7] V. Johnson, *Battery performance models in ADVISOR*, *J. Power Sources* (2002).
- [8] J.N. Harb, *Development and Integration of a Fundamentally Based Battery Model for Low-Emission Vehicle Simulations*, NREL Report, June 1999.