

Introduction to Dualfoil 5.0

Paul Albertus and John Newman

August 14, 2007

Contents

1 Introduction

2 Description of what an “average user” should be able to do

2 Notes on operating dualfoil5.f

3 Input file

4 Output files

1 Introduction

This document serves as an introduction to Dualfoil version 5, and provides information about how to use the program. A series of major modifications to the code made in the period from Summer 2005 to Summer 2007 has made some parts of the previous introduction to dualfoil.f (Dualfoil version 4) outdated. However, the present document builds on the previous introduction (October, 2002), which still may be helpful for the user. We note that while there have been changes to the input file, output file, and main code, all of the functionality of Dualfoil 4 has been retained. That is, Dualfoil 5 has additional features while retaining all the features of Dualfoil 4.2.

Dualfoil.f is a Fortran program intended for the simulation of intercalation batteries, including the lithium-metal, lithium-ion, sodium-ion, and nickel metal-hydride chemistries. The equations and methods used in the program are largely based on the following publications:

Marc Doyle, Thomas F. Fuller, and John Newman, *J. Electrochem. Soc.*, 140 (1993), 1526-1533.

Thomas F. Fuller, Marc Doyle, and John Newman, *J. Electrochem. Soc.*, 141 (1994), 1-10.

Thomas F. Fuller, Marc Doyle, and John Newman, *J. Electrochem. Soc.*, 141 (1994), 982-990.

A description of the use of the model for the NiMH system, as well as the use of the model for treating side reactions, is forthcoming. Future publications will also discuss the use of the model for treating multiple particle sizes.

Previous work has been published on the use of dualfoil.f with two characteristic particle sizes:

Robert Darling and John Newman, *J. Electrochem. Soc.*, 144 (1997), 4201-4208.

Here is a list of the major changes made relative to Dualfoil v. 4.2

- Addition of specified load and power modes. Note that for the specified power mode during discharge, once the specified power equals the maximum power, the program will calculate and use the maximum power.
- Ability to specify an upper and lower cutoff potential for each segment. Also, the specific energy and power are now reported for each operating segment.
- Addition of NiMH materials (MH anode, NiOOH cathode, KOH electrolyte).
- Addition of side reactions and material balances to model gas-phase recombination in the NiMH system. The treatment of the NiMH system also includes a volume correction factor for the presence of headspace in the system, called "vol," and defined as the amount of gas volume in the electrodes and separator divided by the total amount of gas volume in the system.
- Ability to use a variable solid-phase diffusion coefficient in a "pseudo-2d" arrangement. This is useful for modeling systems with a solid-state diffusion coefficient that depends on temperature and/or concentration.
- Use of a new input file (ebar.in) with activation energies for the temperature dependence of the transport properties (solid and electrolyte diffusion, electrolyte conductivity), rate constants, and film resistances.
- Detailed profiles are now written as the program goes along (rather than after the code has finished running) to a separate file, called profiles.out, if the user selects ill=1.
- A problem with the identity/capacity/density of the cathode in the default input file has been corrected.
- The unused subroutine util has been removed.
- An additional equation has been added to improve convergence. This equation carries the potential from the outside of the cell to a point in the interior.
- The method of running constant potential, power, and load has been improved.
- The ability to have up to five particles per electrode has been added.
- A residual mass has been added that is used in the energy balance and accounts for any packaging above the cell level. Search for "residm," which has units of kg/m^2 .
- The open-circuit potential in version 4.0 was printed at the incorrect time step (the value displayed was from the previous, not the current, time step), and this has now been corrected.
- The variables and equations have been given names rather than numbers. This allows the number of equations to be easily changed depending on the number of particles in the system. The user can also specify the number of side reactions to include, which will change the number of equations used by the program.
- The energy balance is now done according to the method given in Rao and Newman, J. Electrochem. Soc., 144 (1997), 2697-2704. The energy balance has been modified so that it will work for multiple particles by defining a pore wall flux for each individual particle.

2 Notes on what an "average user" should be able to do

As the user will quickly notice, dualfoil5.f is a complicated program. Following is a list of what we expect an average user to be able to change, and what an average user would need our assistance to do.

Sample list of tasks an inexperienced user should be able to do, and a brief explanation of how to do these tasks:

- Change the operating mode and other operating parameters (such as current). These changes can be made in the input file just beneath the line that specifies `lcurs`.
- Adjust the cell geometrical parameters, such as thicknesses and porosities. These changes are made in the input file.
- Modify the initial and ambient temperature, and select whether a run is carried out isothermally or nonisothermally. Adjust the activation energies. These changes are made either in the main input file or, in the case of the activation energies, in the “`ebar`” input file.
- Adjust parameters such as the solid-state diffusion coefficients, rate constants for the main reaction, and solid-phase conductivity to explore the effects of different types of resistances. These adjustments can all be made in the main input file. Note that the liquid-phase diffusion coefficient and conductivity are in subroutine `prop`; different solvents and electrolytes are chosen in the main input file.
- Change the materials in the system, including the anode, cathode, and electrolyte. This is done near the bottom of the input file.
- Run the system in impedance mode. This is done by changing “`imp`” from 0 to 1 in the main input file. Note that impedance mode only works with a single particle, isothermal operation, superposition to calculate solid-state diffusion, and no side reactions. Also, the mode should be galvanostatic and the current should be set to zero when running the impedance mode.
- Input and run the system with a variable diffusion coefficient. Change `mvdc1` and/or `mvdc3` to 1, and input the desired function and its derivative for the dependence of the solid-state diffusion coefficient on concentration and temperature. The functions can be found by searching for the phrase “For a true variable solid-phase diffusion coefficient.”
- Add additional materials for which the user has data. The user should change the appropriate value of `nneg`, `nprop`, or `npos` to “Add your own,” and then input the appropriate data in subroutine `ekin` (for a user’s own electrode material) or `nprop` (for a user’s own electrolyte).
- Change the number, size, and fraction of particles in each electrode. For this the user needs to go to the main code and search for “`npa=`” Here, the user should input the number of particles that are desired (from 1 to 5). To use nonuniform particles, search for “For nonuniform particles use this section” and uncomment the code in this section. This part of the code is currently set up to model a two-particle distribution, and the user should be able to follow this example and read the comments in this section to make additional changes.

- (More advanced) Switch from a side reaction framework intended for a NiMH cell to one for a Li-ion cell. Some changes the user may want to make in this case are listed here. Change the phase in which the reaction occurs by changing the volume fractions used in the material balance in equation i7. Change the stoichiometry of the reaction by changing the value of “sf” in equation i7. Change the side reaction kinetics to make them appropriate for a side reaction in a lithium ion system by changing the location of comments in equation i8.

3 Notes on operating Dualfoil 5.0

In this section we discuss the use of the additional features of Dualfoil 5.0. For use of the input file, see the next section. Read this section carefully if you are planning to use the additional features of Dualfoil 5.0, as they are not all compatible with one another.

Notes on operating Dualfoil 5.0:

- There are still some problems with the maximum power mode. It should also be noted that the maximum power calculated by the code depends on the length of the time step.
- The program will not work with the lht=1. The code will only work with lht=0 (the user inputs the heat-transfer coefficient) or lht=2 (isothermal).
- The impedance feature will only work with a single particle size in each electrode, the use of superposition to calculate the solid-phase diffusion, and without side reactions.
- While there can be multiple particle sizes, there is a single value of the film resistance for each electrode.
- The variable-diffusion-coefficient feature will only work with a single particle size.
- The system can be run in a nonisothermal manner with superposition if the activation energies for solid-state diffusion are set to zero.
- It is possible to use side reactions and nonzero values of the capacitances simultaneously.
- The side reaction framework is meant to be general, but is currently set up for simulating the oxygen and hydrogen side reactions occurring in the NiMH system. The material balances and the rate expressions may need to be changed if a different reaction or chemistry is used. For example, for a liquid-phase redox shuttle, the stoichiometry, volume fraction, and kinetic expression would need to be changed. For reactions involving the gas phase, the initial pressures of reacting species can be set by searching for “co2init” and “ch2init,” and the pressure of an inert can be set by searching for “cn2.” Also note that the hydrogen side reaction is set up to allow absorption of hydrogen into the metal hydride material. This required modification of the solid-phase material balance in the negative electrode to allow the flux of hydrogen across the interface due to either the electrochemical insertion reaction or the adsorption/absorption process. The user will need to modify this in order to use the second side reaction if a different phenomenon is being modeled.

- The number of particles is set with the variable “npa.” The default is “npa=1.” If the user chooses a value for npa besides 1, the default is to use particles of the radii specified in the input file and to give them equal areas. For nonuniform particle sizes, uncomment the section that begins with “For nonuniform particles use this section.” The user can input the radius and number fraction of each particle size. The program will then calculate the specific area (m^2 interfacial area / m^3 electrode volume) occupied by each particle size, and the total specific area of all particles. Note that it is the user’s responsibility to ensure that the number fractions of particles add up to 1.

4 Input files

The basic framework for operating Dualfoil 5.0 is the same as for Dualfoil 4, and we refer the reader to the introduction to Dualfoil 4 for those instructions. In this section we discuss the changes to the input file made for Dualfoil 5.0. One important change is that an additional input file has been added, called ebar.in. This file contains activation energies (expressed as E_a/R , with units K) for the appropriate properties in the system. In the version we have posted, we have set all these values to zero. For characteristic values, try 4000 K. Note that the use of the superposition integral is valid only for constant values of the solid-phase diffusion coefficient. In order to use the superposition integral for the solid-phase material balance while operating the cell in a nonisothermal manner, set the activation energies of the solid-phase diffusion coefficients to zero. To use a nonzero activation energy for the solid-phase diffusion coefficient, the user should use the material balance for the variable solid-phase material balance (set mvdc1 and/or mvdc3 to 1, as appropriate).

Following is a list of changes to the main input file, dualfoil5.in.

- n4. For a discretized material balance in the solid-phase, the numbers of nodes in the solid-phase can be specified.
- mvdc1 and mvdc3. These are flags to turn on the variable diffusion coefficient for the negative and positive electrodes, respectively. Note that this feature can be used only with a single particle size, and that the diffusion coefficient can vary with temperature and/or concentration.
- lims. Limit on the number of iterations for convergence for the solid-phase dimension. Note that this is used only if mvdc1 and/or mvdc3 is set to 1.
- epg1, epg2, epg3. The volume fraction of a gas phase in each region of the cell can be specified.
- nside. Flag to specify the number of side reactions in the system. 0 for no side reactions, 1 for one side reaction (this will add two equations: one kinetic expression and one material balance), 2 for two side reaction (this will add four equations: two kinetic expressions and two material balances), and 3 for three side reactions (this will add five equations: three kinetic expressions and two material balances). One of the side-reaction kinetic expressions is not paired with a material balance to account for the case in which there is more than one side

- reaction on the same species (*e.g.*, hydrogen generation and absorption occurring in the same electrode).
- rk_{sa1} and rk_{sc1}. Side-reaction kinetic constants for the first side reaction for the negative and positive electrodes, respectively.
 - rk_{sa2} and rk_{sc2}. Side-reaction kinetic constants for the second side reaction for the negative and positive electrodes, respectively.
 - rk_{sa3} and rk_{sc3}. Side-reaction kinetic constants for the third side reaction for the negative and positive electrodes, respectively.

There has also been a number of inputs removed from the input file.

- vc_{uthi} and vc_{utlo}. The upper and lower cutoff potentials have been moved to the lines where the mode of operation is defined (bottom of the input file).
- dUdT. The entropy term is now located only in subroutine ekin.

5 Output files

There are up to four output files generated by the code. The main output file (dualfoil5.out) reports the most important variables throughout the run, such as time, current, potential, and temperature. The output file profiles.out reports (provided that il1 is set to 1) a detailed profile throughout the cell with the time and spatial resolution chosen by the user (by setting the values of il2 and il3). These profiles used to be written at the bottom of the output file dualfoil5.out, but is now written to a separate file. The output file resistances.out reports the total resistance of the cell, as well as the resistance in each region of the cell. The resistances are calculated from a simple $\Delta V/I$ calculation. For example,

$$\text{Cell Resistance} = \left| \frac{V_{cell} - U_{cell}}{I} \right|$$

Here, V_{cell} is the actual cell voltage (V), U_{cell} is the equilibrium potential of cell (V), and I is the current (A/m²). The final output file, solidprof.out, reports the concentration profile inside a solid particle. This is currently set up to only work with the superposition mode.