Warthog: Coupling Nek5000 Thermal Hydraulics to BISON Fuel Performance through the Giraffe Interface

Kevin J. Dugan
Shane W. D. Hart
Bradley T. Rearden

October 2018

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE US DEPARTMENT OF ENERGY

Reactor and Nuclear Systems Division

WARTHOG: COUPLING NEK5000 THERMAL HYDRAULICS TO BISON FUEL PERFORMANCE THROUGH THE GIRAFFE INTERFACE

Kevin J. Dugan
Shane W. D. Hart
Bradley T. Rearden

Date Published: October 2018

Contents

1. INTRODUCTION

The Nuclear Energy Advanced Modeling and Simulation (NEAMS) program under the US Department of Energy (DOE) consists of three product lines. The Fuels Product Line (FPL), hosted at Idaho National Laboratory (INL), has produced a multiphysics framework with the name Multiphysics Object Oriented Simulation Environment (MOOSE) (Gaston et al. 2015). The FPL has also produced several high-fidelity physics codes including the fuel performance code Bison (Hales et al. 2016). The Reactors Product Line (RPL), hosted at Argonne National Laboratory (Argonne), has developed the high-order spectral-element Computational Fluid Dynamics (CFD) code Nek5000 (Fischer et al. 2015). The Integration Product Line (IPL), hosted at Oak Ridge National Laboratory (ORNL), is tasked with ensuring tools from the other product lines can be coupled together so that users may perform high-fidelity multiphysics simulations.

The Warthog project (Dugan, Hart, and Rearden 2018), under the IPL, is serving as a key component in connecting tools produced under the RPL to the MOOSE framework. Warthog has been shown to successfully couple the neutronics code PROTEUS (Shemon, Smith, and Lee 2016) with Bison. The Giraffe application (Merzari and Rahaman 2017) has been developed in an effort to provide an interface between Nek5000 and the MOOSE framework. This report describes the integration of the Giraffe application into Warthog to provide a coupling mechanism between Nek5000 and Bison. Giraffe provides a way to control calculations in Nek5000 and transfer solutions to and from Nek5000. This report examines the solution transfer mechanism in Giraffe, describes the architectural changes performed to integrate Giraffe into Warthog, and shows an example problem which highlights the coupling mechanism between Nek5000 and Bison.

2. GIRAFFE SOLUTION TRANSFERS

Giraffe transfers solutions between Nek5000 and MOOSE applications by constructing a polynomial expansion of the solution over a bounding surface. In the cases relevant to NEAMS, the bounding surface is typically a cylinder representing the clad outer surface in contact with a flowing fluid. Giraffe expands both the temperature and the heat flux over the clad surface. The heat flux typically supplies a boundary condition to the CFD simulation, while the temperature provides a boundary condition for the heat transfer simulation within the fuel. The expansion is performed using a Legendre-Fourier transform

$$F(x, y, z) = \sum_j \sum_k f_{jk} \, m_j(\theta) l_k(\eta), \tag{1}$$

where the coefficients f_{jk} are computed from orthogonality relations, and the functions $m_j(\theta)$ and $l_k(\eta)$ are Fourier-cosine and Legendre functions, respectively. The domain for the Legendre functions is $[-1, 1]$, while the domain for Fourier expansions is $[-\pi, \pi]$. Hence, the physical Cartesian domain of the simulation must be mapped to a domain where these expansions are valid. The normalized parameters $\{\theta, \eta\}$ relate to the Cartesian coordinates through the relations

$$\eta = \frac{2(z - z_0)}{L} - 1,$$
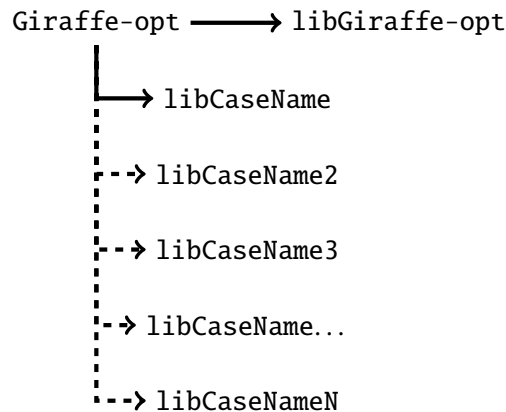
$$\theta = \text{arctan2}\,(y, x)\,, \tag{2}$$

where z_0 and L are the minimum value for z and the total axial length, respectively. The arctan2 function is an extension of the arctan function which accounts for the sign of the second argument. The arctan2 function uses the entire azimuthal angle range of $[-\pi, \pi]$ much like the Arg function from complex analysis.

The radius is assumed constant during the simulation and is hidden within the expansion coefficients. Because of the constant radius assumption, this transfer mechanism is not suitable for simulations where the clad outer surface deforms (i.e., after pellet-clad interaction). The use of Fourier-cosine functions in angle implicitly enforces plane symmetry in angle; an example of the weakness inherent in this implementation is explored in Section 4.

Only the expansion coefficients are transferred between Nek5000 and MOOSE apps, which allows for constant time complexity of the solution transfer. This design for solution transfers means that the transfers will take the same time regardless of the level of mesh refinement but sacrifices the generality of a Finite-Element Mesh Transfer.

3. REFACTORING GIRAFFE

The original implementation for Giraffe was designed to operate as a master app, controlling the MOOSE modules connected to it. However, this design made it difficult to use Giraffe as a Multiapp within Warthog. Several aspects of Giraffe were redesigned to transition Giraffe to function as a Multiapp driven by another MOOSE-app. A secondary aspect of this redesign made Giraffe a little less cumbersome by extracting the architectural pieces that were not example-dependent. These pieces were combined into a library that could be linked by the executable. Nek5000 has a unique design which requires the user to compile input parameters and the original implementation of Giraffe reflected this design. The previous implementation required Giraffe to be recompiled from scratch when any changes were made to the example problem input; the redesign only requires recompiling those pieces that are problem-dependent.

```
Giraffe-opt ──────⟶ libGiraffe-opt
      │
      └──⟶ libCaseName
      ¦
      ┊--⟶ libCaseName2
      ¦
      ┊--⟶ libCaseName3
      ¦
      ┊-⟶ libCaseName...
      ¦
      ┊--⟶ libCaseNameN
```

Figure 1: Linking structure for redesigned Giraffe app.

Figure 1 shows the linking scheme for the new Giraffe executable. In the figure, `Giraffe-opt` designates the executable, `libGiraffe-opt` designates the library of Giraffe capabilities that are not case dependent, and each `libCaseName` designates a library with case-specific information. When switching between different Giraffe cases, the executable may be re-linked to the appropriate case library. Additionally, when case-specific information is changed, only the case-specific library needs to be recompiled and linked to the executable; the `libGiraffe-opt` library may be left unchanged. This capability makes Giraffe more

user-friendly considering that changing certain parameters within a Nek5000 example input requires the user to recompile Fortran source code.

4. COUPLING EXAMPLE

The new coupling capabilities gained by integrating Giraffe into Warthog are demonstrated by a small example in this section. The chosen example calculates the effect of a slug of hot water entering a fuel channel at zero power; the fluid velocity is a uniform 1.0 cm/s. The channel is initially at a uniform temperature of 300 K when a hot slug of water at 500 K and 0.2 cm thick enters the channel. A zero power transient was chosen to highlight the solution transfers between the Nek5000 and Bison simulation without losing certain solution details due to the potentially large gradients present at higher powers.

(a) Time = 0.02s   (b) Time = 0.10s

(c) Time = 0.20s   (d) Time = 0.50s
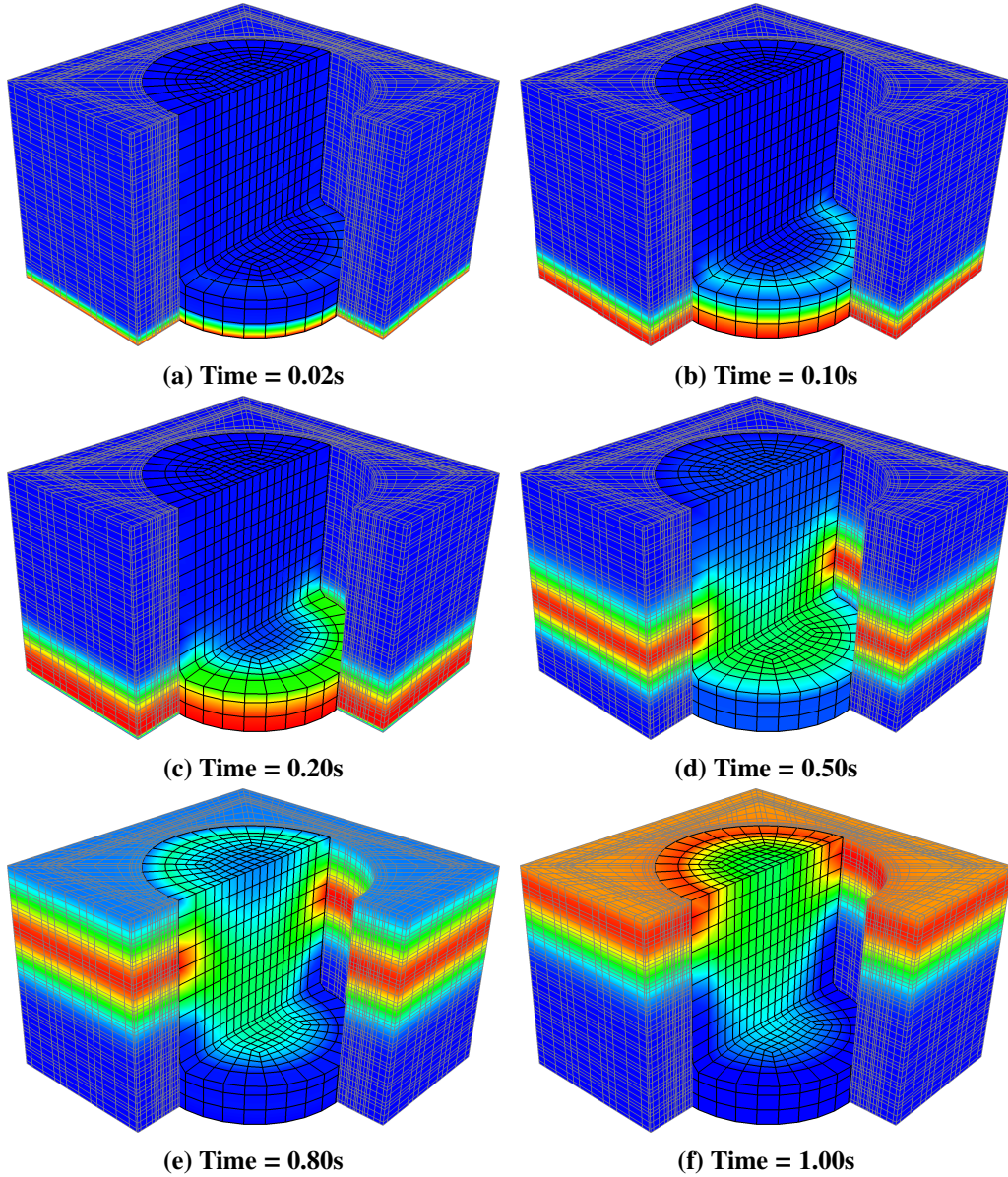
(e) Time = 0.80s   (f) Time = 1.00s

Figure 2: Temperature distribution for slug of hot water entering channel.

Figure 2 shows the evolution of this hot water slug as it moves through the channel. The simulation is run until 1.0 s with a uniform time step of 0.001 s. The figures show a cutaway view of the coolant channel and the fuel pin; the fuel pin (fuel pellet with clad and no gas gap) has a solid black mesh, while the coolant channel has a grey mesh. The temperatures range from 300 K (blue) to 500 K (red) in Figure 2.

It can be seen in Figures 2c & 2d especially that the temperature along the clad outer surface follows the temperature of the hot water slug axially. The figures also show that there is a time delay between the center of the fuel pellet *feeling* the effects of the transient; this effect is evident in the temperature wake that appears in Figure 2f.
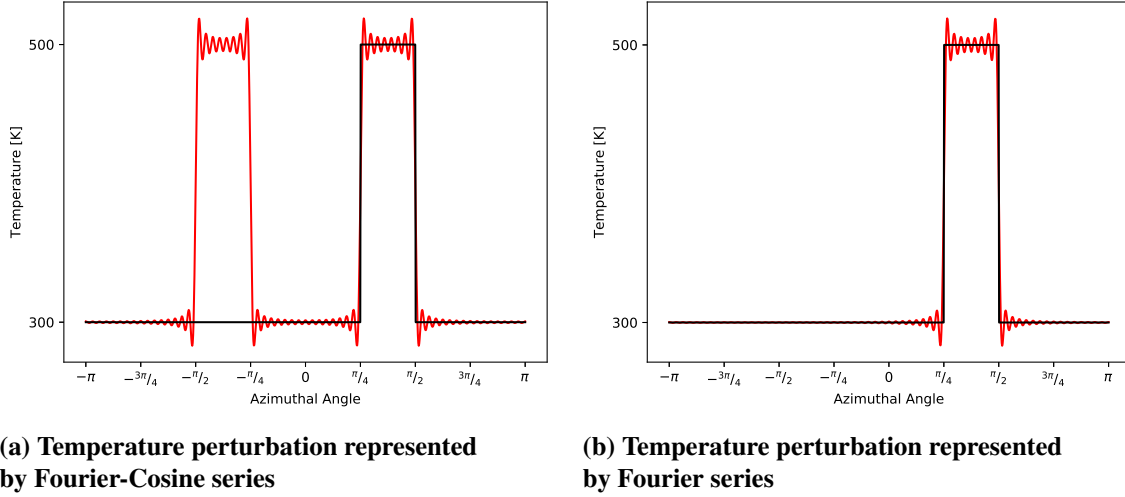
The example problem depicted in Figure 2 fits nicely into the assumptions inherent with Giraffe's implementation, but these assumptions can pose problems when the temperature profile within the coolant channel is not azimuthally symmetric. Figure 3 shows such an example where the temperature profile within the coolant channel is not symmetric in the azimuthal angle. In this situation, the hot water slug is confined to an eighth of the coolant channel. One would expect the temperature within the fuel pellet to radiate out from the hot spot on the clad surface without developing a hot spot on the clad surface elsewhere. However, the temperature profile in Figure 3 for the fuel rod displays a hot spot forming on the opposite side of the fuel where there is no heat entering from the coolant. This, of course, is nonphysical behavior and simply an artifact of the way the solution is decomposed on the clad surface.
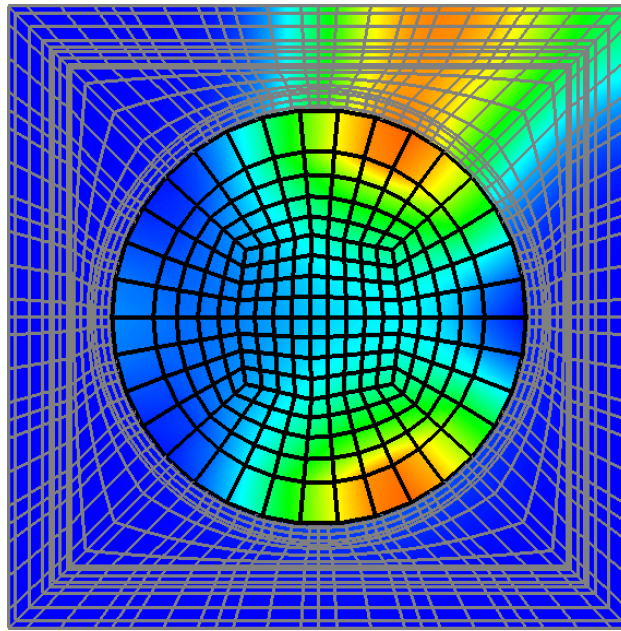
Figure 3: Breakdown of function expansion implementation.

A simple remedy to this behavior is to make use of a full Fourier expansion in the azimuthal angle. The current implementation of Giraffe expands the solution on the clad surface using a Fourier-Cosine expansion which assumes even-symmetry. Moving to this type of expansion will increase the number of coefficients needed to be transferred between Nek5000 and MOOSE apps but will greatly increase the generality of Giraffe's capabilities. Figure 4 compares the two expansion methods applied to a non-symmetric pulse; both figures use 64 terms in the Fourier expansion. Figure 4a shows the artificial pulse produced when using only cosine expansion terms; this artificial pulse is equivalent to the temperature distribution shown in Figure 3. Introducing the sine terms to the Fourier expansion in Figure 4b removes the artificial pulse and aligns closely with the actual temperature distribution.

(a) Temperature perturbation represented by Fourier-Cosine series

(b) Temperature perturbation represented by Fourier series

Figure 4: Comparing a full Fourier series to a Fourier-Cosine series to represent temperature.

5. CONCLUSION

This report details the work completed toward integrating the Giraffe app into the Warthog app. The capabilities implemented in Giraffe are desirable for performing simulations with Warthog. However, Giraffe required several architectural changes before it could be used as a Multiapp within Warthog. A key benefit in the redesign of Giraffe's architecture came in the form of a more problem-independent build. The previous implementation required the entire Giraffe app, including the underlying Nek5000 source code, to be recompiled from scratch when changes were made to the input of example problems; the current implementation only requires a problem-specific library to be recompiled when changes to the input occur.

The solution transfer mechanism was examined and analyzed. Transferring expansion coefficients between Nek5000 and MOOSE applications does provide for a simple and efficient solution transfer but sacrifices some generalities available in more robust solution transfer methods. A key weakness in the current implementation was examined where asymmetries cause nonphysical behaviors. A remedy for this situation is proposed by extending the expansion terms used for mapping the solution in azimuthal angle.

Finally, the coupling mechanism between Nek5000 and Bison was demonstrated using a simple case where a slug of hot water enters a coolant channel at zero power. The example problem demonstrates that the coupling between the CFD simulation and the heat transfer simulation within the fuel rod behaves realistically.

5

6. REFERENCES

Dugan, Kevin J., S. W. D. Hart, and Bradley T. Rearden. 2018. *Warthog Framework Update: Depletion Capabilities.* Technical report ORNL/TM-2018/490. Oak Ridge National Laboratory, March.

Fischer, P., J. Lottes, S. Kerkemeier, O. Marin, K. Heisey, A. Obabko, E. Merzari, and Y. Peet. 2015. *Nek5000: User's Manual.* Technical report ANL/MCS-TM-351. Argonne National Laboratory.

Gaston, Derek R., Cody J. Permann, John W. Peterson, Andrew E. Slaughter, David Andrs, Yaqi Wang, Michael P. Short, et al. 2015. "Physics-based multiscale coupling for full core nuclear reactor simulation." Multi-Physics Modelling of LWR Static and Transient Behaviour, *Annals of Nuclear Energy* 84:45–54. ISSN: 0306-4549. doi:https://doi.org/10.1016/j.anucene.2014.09.060. http://www.sciencedirect.com/science/article/pii/S030645491400543X.

Hales, J. D., R. L. Williamson, S. R. Novascone, G. Pastore, B. W. Spencer, et al. 2016. *BISON Theory Manual: The Equations Behind Nuclear Fuel Analysis.* Technical report INL/EXT-13-29930. Idaho National Laboratory.

Merzari, Elia, and Ron Rahaman. 2017. "Giraffe: A MOOSE-wrapped app for Nek5000." Personal Communication, September.

Shemon, E. R., M. A. Smith, and C. Lee. 2016. *PROTEUS-SN User Manual.* Technical report ANL/NE-14/6 (Rev 3.0). Argonne National Laboratory, February.