Updating a Model from NPSS v2.3.0.1 to v2.4.1

Presented by Wolverine Ventures

NPSS™ software and related documentation is export controlled with an Export Control Classification Number(ECCN)of 9D991, controlled for Anti-Terrorism reasons, under U.S. Export Administration Regulations 15 CFR 730-774. It may not be transferred to a country checked under anti-terrorism on the Commerce Country Chart structure or to foreign nationals of those countries in the U.S. or abroad without first obtaining a license from the Bureau of Industry and Security, United States Department of Commerce. Violations are punishable by fine, imprisonment, or both.

Disclaimer: This documentation is provided "as is" without any warranty of any kind, either expressed, implied, or statutory. In no event shall Wolverine Ventures Incorporated, be held liable for any damages, including, but not limited to, direct, indirect, special, or consequential damage.

Contents

- How the Compressor has changed from NPSS v2.3.0.1 to v2.4.1
- How the Turbine has changed from NPSS v2.3.0.1 to v2.4.1
- How setting a thermo package to CEA has changed from NPSS v2.3.0.1 to v2.4.1

Backwards Compatibility

- With the release of NPSS version 2.4.1 the user may encounter some errors if he/she tries to run a model that was written and runs in NPSS version 2.3.0.1
- While NPSS version 2.4.1 release brings many enhancements and bugs fixes, most of these features will not impede the execution of a model written in NPSS version 2.3.0.1
  - See file "ReleaseNotes.txt" in the NPSS root directory for a list of changes from v2.3.0.1 to v2.4.1
- A couple items that may cause your v2.3.0.1 mode to fail in v2.4.1 are:
  - Updated Compressor element and subelement architecture
  - Updated Turbine element and subelement architecture
  - Updated method for setting the thermo package to CEA

Updated Compressor Element

- If your v2.3.0.1 model uses the Compressor element from the NPSS library but does not make use of the S_map socket, your Compressor element should function the same as before
- Instantiation, and inputs for design pressure ratio and efficiency of a Compressor element without maps remain unchanged:

```
Element Compressor Cmp {

        PRdes  = 10.0; // design point pressure ratio
        effDes= 0.85; // design point efficiency

} // End Cmp
```

Updated Compressor Element

- If your v2.3.0.1 model uses the Compressor element from the NPSS library AND also makes use of the CompressorMap and CompressorEfficiencyMap subelements, your Compressor element instantiation and map files must be updated
- PRdes and effDes are no longer input in the S_map socket subelement; they are always set in the Compressor element

v2.3.0.1

```
Element Compressor Cmp {

        #include "hpcE3.map"; // compressor performance map

        // Set compressor design point values in S_map subelement
        S_map{
                PRdes  = 10.0; // design point pressure ratio
                effDes = 0.85; // design point efficiency
        }
} // end Cmp
```

v2.4.1

```
Element Compressor Cmp {

        PRdes  = 10.0; // design point pressure ratio
        effDes = 0.85; // design point efficiency

        #include "hpcE3.map"; // compressor performance map

} // end Cmp
```

Updated Compressor Subelements

- NPSS v2.3.0.1 came with a subelement named CompressorMap which was designed to plug into the Compressor S_map socket. It also came with a subelement named CompressorEfficiencyMap which was designed to plug into the CompressorMap S_eff socket.
- NPSS v2.4.1 no longer comes with the CompressorMap and CompressorEfficiencyMap subelements. Instead, it comes with a CompressorRlineMap subelement that is designed to plug into the Compressor S_map socket.
- CompressorRlineMap is essentially a combination of the old CompressorMap and CompressorEfficiencyMap subelements. The name implies that it is designed to read maps that are R-line based, just like the maps used with the NPSS v2.3.0.1 example turbofan model used.
- The example turbofan model in NPSS v2.3.0.1 came with the compressor map files hpcE3.map, lpcE3.map, and fanE3.map. The example turbofan model in NPSS v2.4.1 also comes with these three map files, but since the compressor subelement architecture has changed, so have these maps files
- Three new subelement variables have been created to represent the unscaled map's design point
  - alphaMapDes – unscaled map design point vane angle
  - NcMapDes – unscaled map design point corrected speed
  - RlineMapDes – unscaled map design point Rline
  - Note: The v2.3.0.1 sublement used alpha, NcDes, and RlineMap to set the unscaled map's design point

Updated Compressor map file

- Comparison of v2.3.0.1 and v2.4.1 map files (example is hpcE3.map)
  - Notice that CompressorMap has been renamed, CompressorEfficiencyMap no longer exists, and unscaled map design point variables have been renamed

v2.3.0.1

```
Subelement CompressorMap S_map {

  // Unscaled map design point
  RlineMap = 2.0; // Max eff point
  NcDes    = 1.0; // Design speed (100%)

  Subelement CompressorEfficiencyMap S_eff{

    RlineStall= 1.0; // stall line

    TB_Wc(real alpha, real SPED, real R){
        Corrected flow table contents here
    }

    TB_PR(real alpha, real SPED, real R) {
        Pressure ratio table contents here
    }

    TB_eff(real alpha, real SPED, real R) {
        Efficiency table contents here
    }

  } // end S_eff

} // end S_map
```

v2.4.1

```
Subelement CompressorRlineMap S_map {

  // Unscaled map design point
  alphaMapDes = 0.0;  // vane angle
  RlineMapDes = 2.0;  // Max eff point
  NcMapDes    = 1.0;  // Design speed (100%)
  RlineStall  = 1.0 ; // stall line

  TB_Wc(real alpha, real SPED, real R){
        Corrected flow table contents here
  }

  TB_PR(real alpha, real SPED, real R) {
        Pressure ratio table contents here
  }

  TB_eff(real alpha, real SPED, real R) {
        Efficiency table contents here
  }

} // end S_map
```

Updated Compressor Solver

- The v2.3.0.1 CompressorMap sublement contained a solver balance that activated during off-design mode. The v2.4.1 CompressorRline subelement contains the exact same solver variables which involve the exact same compressor variables.

```
Independent ind_RlineMap {
        varName = "RlineMap";
        description = "Compressor operating point independent";
}

Dependent dep_errWc {
        eq_lhs = "Wc";
        eq_rhs = "WcCalc";
        description = "Flow error dependent";
}
```

Updated Turbine Element

- If your v2.3.0.1 model uses the Turbine element from the NPSS library but does not make use of the S_map socket, your Turbine element will require one minor change
- The v2.3.0.1 Turbine element used eff as the user-specified efficiency, but v2.4.1 uses effDes as the user-specified efficiency
- Instantiation of a Turbine element without maps:

v2.3.0.1

```
Element Turbine Trb {

        PRbase  = 5.0; // pressure ratio
        eff     = 0.85; // design point efficiency

}
```

v2.4.1

```
Element Turbine Trb {

        PRbase  = 5.0; // pressure ratio
        effDes  = 0.85; // design point efficiency

}
```

Updated Turbine Element

- If your v2.3.0.1 model uses the Turbine element from the NPSS library AND also makes use of the TurbineNeppMap and TurbineEfficiencyMap subelements, your Turbine element instantiation and map files must be updated
- parmMap and effDes are no longer input in the S_map socket subelement; they are always set in the Turbine element as PRbase and effDes, respectively

v2.3.0.1

```
Element Turbine Trb {

          #include "hptE3.map"; // turbine performance map

          // Set turbine design point values in S_map subelement
          S_map{
                    parmMap  = 5.0;  // pressure ratio initial guess
                    effDes   = 0.85; // design point efficiency
          }
} // end Trb
```

v2.4.1

```
Element Turbine Trb {

          PRbase   = 5.0;  // pressure ratio initial guess
          effDes   = 0.85; // design point efficiency

          #include "hptE3.map"; // turbine performance map

} // end Trb
```

Updated Turbine Subelements

- NPSS v2.3.0.1 came with a subelement named TurbineNeppMap which was designed to plug into the Turbine S_map socket. It also came with a subelement named TurbineEfficiencyMap which was designed to plug into the TurbineNepprMap S_eff socket.
- NPSS v2.4.1 no longer comes with the TurbineNeppMap and TurbineEfficiencyMap subelements. Instead, it comes with a TurbinePRmap subelement that is designed to plug into the Turbine S_map socket.
- TurbinePRmap is essentially a combination of the old TurbineNeppMap and TurbineEfficiencyMap subelements. The name implies that it is designed to read maps that are PR-based, similar to the maps used with the NPSS v2.3.0.1 example turbofan model used.
- TurbinePRmap is designed to read maps with pressure ratio and corrected speed as independent parameters, whereas the old TurbineNeppMap and TurbineEfficiencyMap were design to deal with maps with pressure ratio , corrected speed, and geometry as independent parameters. The TurbinePRmap subelement does not contain a geometry parameter
- The example turbofan model in NPSS v2.3.0.1 came with the turbine map files hptE3.map and lptE3.map. The example turbofan model in NPSS v2.4.1 also comes with these two map files, but since the turbine subelement architecture has changed, so have these maps files
- Two new subelement variables have been created to represent the unscaled map's design point
  - PRMapDes – unscaled map design point pressure ratio
  - NpMapDes – unscaled map design point corrected speed
  - Note: The v2.3.0.1 sublement used parmMapDes and parmNcDes to set the unscaled map's design point

Updated Turbine map file

- Comparison of v2.3.0.1 and v2.4.1 map files (example is hptE3.map)
  - Notice that TurbineNeppMap has been renamed, TurbineEfficiencyMap no longer exists, unscaled map design point variables have been renamed, and "geom" is no longer a parameter

v2.3.0.1

```
Subelement TurbineNeppMap S_map {

  // Unscaled map design point
  parmGeomDes  = 76.;
  parmNcDes    = 100.; // design speed (100%)
  parmMapDes   = 4.975; // design pressure ratio

  Subelement TurbineEfficiencyMap S_eff{

    TB_Wp (real BETA, real SPED, real PR){
        Corrected flow table contents here
    }

    TB_eff (real BETA, real SPED, real PR) {
        Efficiency table contents here
    }

  } // end S_eff

} // end S_map
```

v2.4.1

```
Subelement TurbinePRmap S_map {

  // Unscaled map design point
  NpMapDes    = 100.; // design speed (100%)
  PRmapDes    = 4.975; // design pressure ratio

  TB_Wp(real SPED, real PR){
        Corrected flow table contents here
  }

  TB_eff (real SPED, real PR) {
        Efficiency table contents here
  }

} // end S_map
```

Updated Turbine Solver

- The v2.3.0.1 TurbineNeppMap sublement contained a solver balance that activated ind_parmMap (pressure ratio) during design mode, and activated ind_parmMap (pressure ratio) and dep_errWp

```
Independent ind_PRbase {
    varName = "parmMap";
    description = "Turbine operating point independent";
}

Dependent dep_errWp {
    eq_lhs = "Wp";
    eq_rhs = "WpCalc";
    description = "Corrected flow error dependent";
}
```

- The v2.4.1 TurbinePRmap subelement behaves similarly, except the solver independent is now named ind_PRbase and it varies PRbase. The solver dependent dep_errWp is exactly the same as v2.3.0.1

```
Independent ind_PRbase {
    varName = "PRbase";
    description = "Turbine operating point pressure ratio,
                   before audit factors are applied.";
}
```

Setting Thermo Package to CEA

- If your model uses the CEA thermo package, then you will have to update the call to the setThermoPackage function
- The CEA thermodynamic property package behaves the same as it did in v2.3.0.1, just the function call has changed
- The user must specify the files to use for thermodynamic properties and transport properties using the second and third arguments in the setThermoPackage function
  - Note that example files thermo.inp and trans.inp have already been supplied for you and are located in the InterpIncludes directory

V2.3.0.1

```
setThermoPackage("CEA", "Air", "H2O");
```

V2.4.1

```
setThermoPackage("CEA", "thermo.inp", "trans.inp", "Air", "H2O");
```