Context Model C++ Class Library

1. Summary

This appendix points to a C++ class library of nominal and extreme climate-based environmental models drawn from the AR 70-38 and ASHRAE specifications.

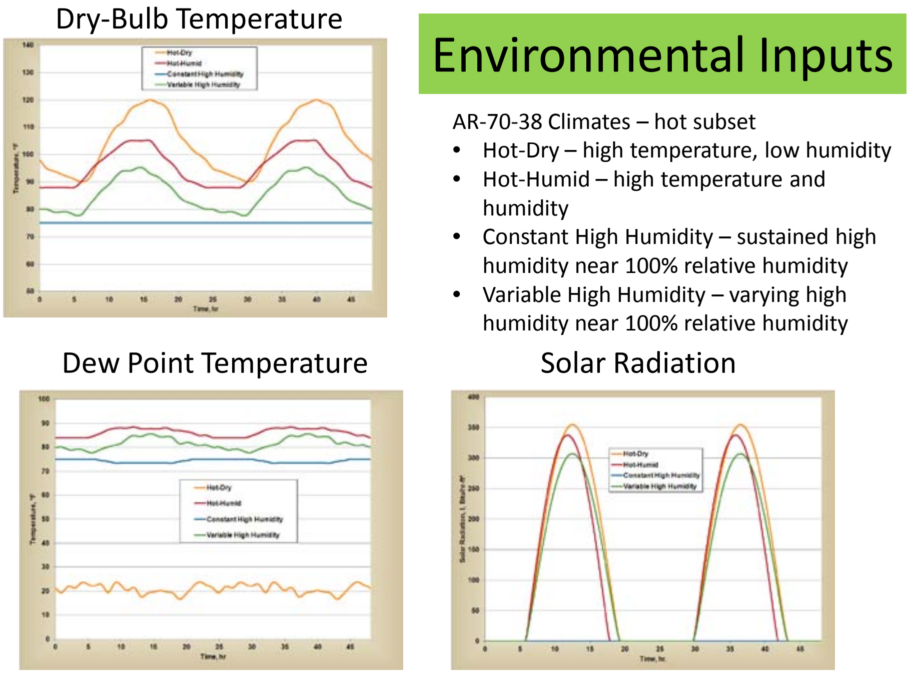


Figure 1: Nominal climate conditions specified in AR-70-38

1.1 Technical Challenges

Synthesize the contents of the specifications to useful library functions. For example the necessary nominal conditions for a thermal model include: temperature, solar radiation, wind speed, and humidity. Optional nominal conditions include: pressure or altitude, contaminants and breakdown of solar radiation into terrain reflection, direct solar (specular, directional), ambient (specular, diffuse), and effective sky temperatures.

1.2 General Methodology

From the specification documents, information in the form of physical constants, tabular data, and algorithms were organized into a C++ class library. Examples and test cases were developed to support usage scenarios. For example, the representation of a thermally loaded vehicle interior which resides within the context of an exterior climate of nominal conditions is shown below:

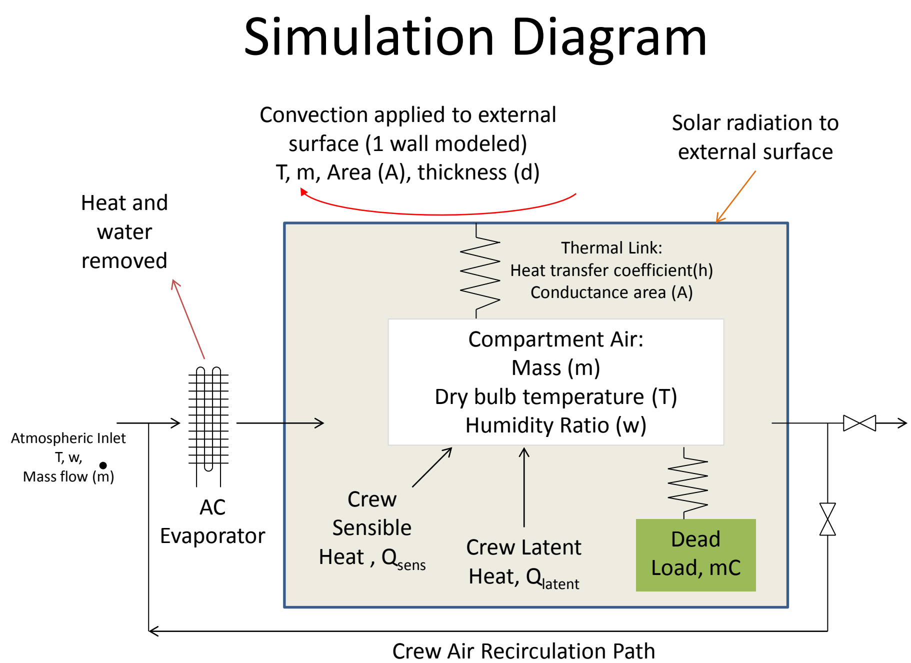


Figure 2: A simulation infrastructure using external contexts

Invoked as stimulus to a thermal solver, the nominal conditions result is shown below

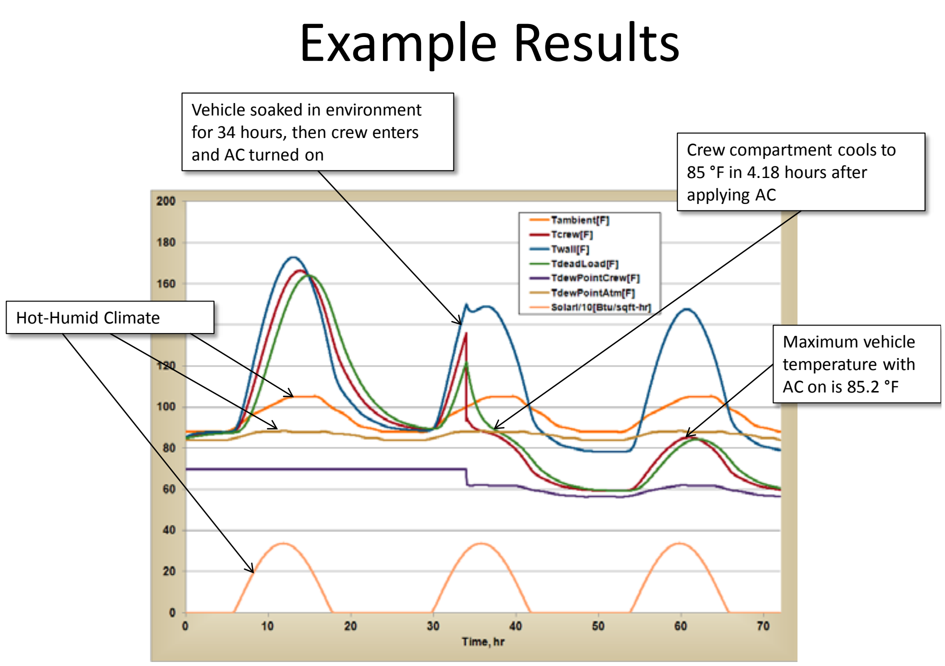


Figure 3: Typical results from applying a nominal diurnal context model.

1.3 Technical Results

The following is a list of C++ classes we have provided:

Climate and Environment Classes

- Class SeaWaterTPProps accesses the thermophysical property tables for salt water as a function of temperature and salinity. See class_SeaWaterTPProps.docx
- Class SolarLoad_ASHRAE encapsulates the ASHRAE Clear Day Model for solar loads. See class_solarLoad_ASHRAE.doc
- Class SolarLoad_ASHRAE2009 is similar to SolarLoad_ASHRAE (see class_solarLoad_ASHRAE.doc) except that it uses the 2009 ASHRAE
- Class ThermalNet models the behavior of a network of thermally connected thermal masses. (see class_ThermalNet.doc)
- Class Climate_AR7038 encapsulates the diurnal cycles of the extreme climatic conditions outlined in Army Regulation 70-38 (AR 70-38). AR 70-38 enumerates 8 climate design types. (see class_Climate_AR7038.doc)
- Class MoistAirMixture approximates the thermodynamic state of a moist air mixture by assuming a binary non-reacting mixture of dry air and water vapor (see class_moistAirMixture.doc)
- Namespace HVAC0 contains auxiliary HVAC functions (see namespace_HVAC.doc)
- Namespace AR7038Environment captures the environment data expressed by Army Regulation AR 70-38 that is not specifically associated with one of the 8 climate design types enumerated in the document and is not covered by class Climate_AR7038
- Class ExplosiveCharge gives the general characteristics of an unconfined hemispherical surface burst of an explosive charge at sea level (see class_ExplosiveCharge.doc).
- Class UnderwaterBlast gives the peak blast pressure, characteristic time, impulse and energy flux density for a spherical explosive charge detonated under water (see underwaterBlast.doc).

Environment Example

Namespace AR7038Environment represents the portions of the AR-70-38 standard that are not categorized in the 8 climate design types. Within namespace AR7038Environment are nested namespaces which group data that address sand and dust particle sizes. An overview of the nesting is given by the table below.

SandDust – particle sizes	Aircraft – particle size, density near aircraft, eg helicopter downwash
	SurfaceVehicle – particle size, density near surface vehicle

	Natural – particle size, density in natural surroundings	FineParticleTesting – typical particle size for testing equipment that is sensitive to fine particles
		LargeParticleTesting – typical particle size for testing equipment that is sensitive to large particles

Use of this namespace allows access to amount and size information about particles for each of these environments. The information is available for use programmatically as illustrated in the following example:

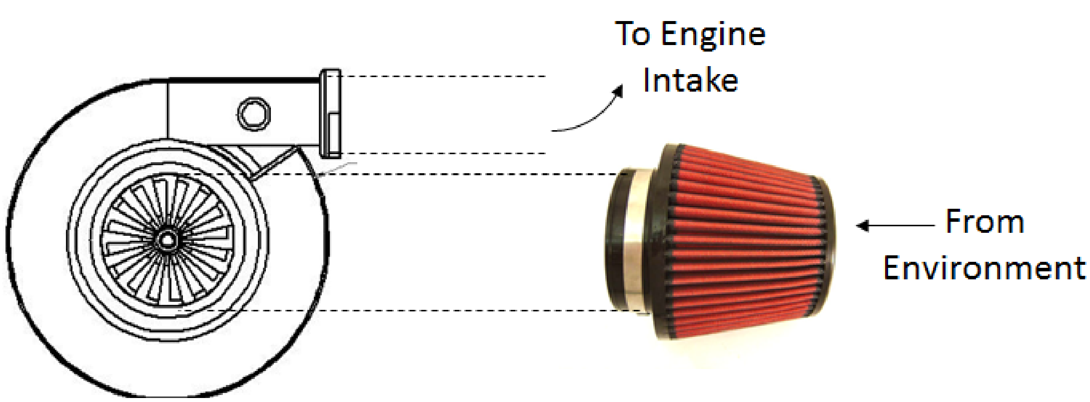
Code Example :

Write the particle density and typical minimum and maximum diameters for environments near operating surface vehicles.

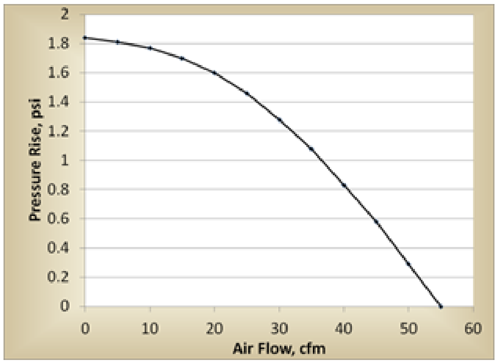
```
#include <iostream>
#include "AR7038environ.h"

int main() {
std::cout << AR7038Environment::SandDust::SurfaceVehicle::rho << '\t'
  << AR7038Environment::SandDust::SurfaceVehicle::diameter_min_typical << '\t';
  << AR7038Environment::SandDust::SurfaceVehicle::diameter_max_typical << '\t';
  return 0;
}
```

Example of Usage for Vehicle Design and Evaluation :

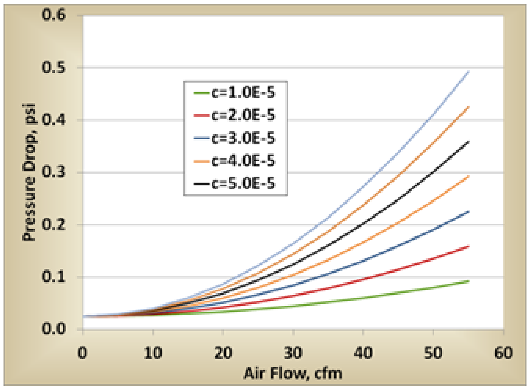
Determine the increasing pressure drop of a filter due to the buildup of particles. In this example, an intake air filter for a vehicle engine is considered. The notional system, illustrated in the figure below, consists of an intake blower such as a superchager or turbocharger and an inlet filter. The system duct work, fittings, and inlets and outlets are modeled as having zero resistance.



The blower is characterized by a pressure output vs. flow curve, which gives the relationship of decreasing pressure for a rising air flow. The filter is characterized by a series of pressure drop vs. flow curves for different ambient air dust concentrations. These curves shift upwards as particulates build up in the media. Not shown or modeled in this example is the filter performance relative to the particle size. The blower curve and filter pressure drop curves are illustrated in the figures below.

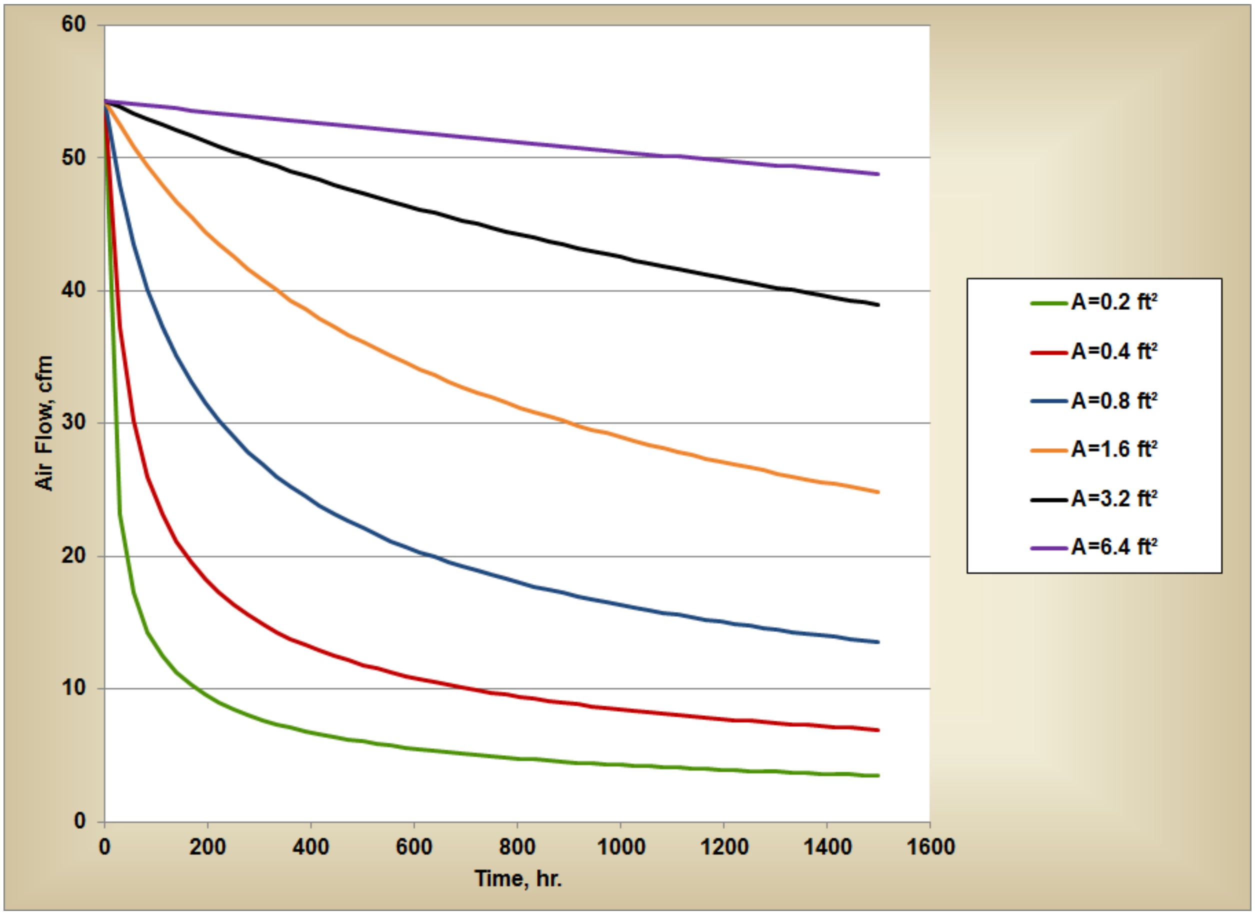


Blower Curve



Filter pressure drop for varying dust concentration

The (quasi) steady state operating point of the system will be the intersection of the blower curve and the filter pressure drop curves. Over time as the filter media accumulates a dust cake, the filter pressure drop curve will shift upwards and the operating point will equilibrate at a higher pressure drop and lower flow rate. The results of this simulation are shown in the figure below for a variety of filter areas.



The C++ code for this example can be found in the non-ITAR svn repository at

\svn\Models\Composite\FluidThermal\test\IntakeFilter

The C++ library code containing the AR-70-38 for dust and other environmental factors is located in the non-ITAR svn repository at

\svn\Models\Composite\FluidThermal\src

1.4 Important Findings and Conclusions

Used as libraries to run examples.

1.5 Implications for Further Research

