A web based program to visualize the transport and thermodynamic properties of thermal plasma

Ambili Sreekumar\(^1\), T.K. Thiyagarajan\(^2\), Vijayalakshmi Ravi\(^1\) and P.V.A Padmanabhan\(^2\)

\(^1\) SIES College of Arts, Science & Commerce, Sector V, Nerul, Navi Mumbai 400 706
\(^2\) Laser & Plasma Technology Division, Bhabha Atomic Research Centre Trombay, Mumbai 400 085

Abstract. The data of transport properties like viscosity, thermal conductivity and electrical conductivity of high temperature gases or gas mixtures used in thermal plasma devices, as a function of temperature is very much essential for the process modeling and theoretical simulation. Similarly the temperature variation of thermodynamic properties like density, specific heat and enthalpy also should be known. Calculated values of these properties of the gases or gas mixtures specific to thermal plasma devices are available as a function of temperature in tabular form. To share the knowledge about these values of properties, web based package of codes and database documents are developed. The modules present in the package works in client-server environment, where the inter communication is through HTTP protocol. The developed package is tested with Apache tomcat web container. The details of the code development and working procedure are described in this paper.

1. Introduction

The transport and the thermodynamic properties of high temperature gases and gas mixtures are indispensable inputs in numerical simulation of plasma processes. The reliability of the simulations is dependent on accurate values for the transport and thermodynamic properties being used. Although it is possible to estimate many of these properties experimentally, calculation remains the most reliable method of determining the required property. The thermodynamic properties required for the modeling of plasma process are density, specific heat and enthalpy. The transport properties for the same requirement are viscosity, thermal conductivity and electrical conductivity. The gases like Argon, Nitrogen, Oxygen, Hydrogen, Helium and gas mixtures like Air, Argon-Air, Argon-Nitrogen, Argon-Oxygen, Argon-Hydrogen, Argon-Helium are used in many important thermal plasma devices, such as plasma spraying, plasma reactors, plasma cutting and welding. Transport and thermodynamic properties are calculated for the above gases and gas mixtures and the tabulated data is available\(^{[1-4]}\) over the temperature range from 300-30000K at a pressure of 101.3kPa. It is reported that the calculations of these property values assume thermodynamic equilibrium.

It is desirable to share the knowledge of these properties through Client-Server environment using the advanced web technology. Users from client side request the properties of selected gas mixtures through an HTML form and this request will be processed in a remote server connected to the client using network/internet. Then the server sends the required information either in the form of graphs in GIF format or ASCI tables to the client using HTTP connection. The present work attempts to create such a Client-Server environment to access and visualize the transport and thermodynamic
properties of high temperature gases to the thermal plasma technologists and thermal plasma user’s community.

As mentioned already, the properties data which are available for each gases and gas mixtures in the form of tabulated data over the temperature range 300-30000K, are compacted by dividing the range into different temperature segments and in each segment the properties data are fitted into cubic polynomial function form and storing only the coefficients of cubic polynomial coefficients. A compact database is created for the transport and thermodynamic properties using the cubic polynomial coefficients. Using MVC architecture described for the server side JAVA programming, a package of computer codes is developed that consists of servlets and Java Server Pages (JSP) in addition to java scripts and HTML forms. The JSPs and servlets of this package are used to get the information from user, to access the data from database, create the graphs in GIF format or create the properties table and transmit the graphs/table to the client. This package of web based program modules, is tested in Apache Tomcat which is a servlet/JSP container provided by the Apache Software Foundation group.

2. Development of web based program package using Java Enterprise Edition

Now a days, web based program modules are developed as a software package using Java programming technologies established for server. These web based program modules are executed in a server (remote computer connected through LAN/Internet) and communicate to the general-purpose client computer. User interacts with the forms displayed in the browser running in the client and the browser communicates the user’s query to the server through HTTP protocol. This process of communicating the user’s query to the server by the client is termed as ‘request’. The server responds to the client’s request by executing the appropriate modules of the deployed package in the server and communicates the required information to the client browser through HTTP protocol. This process of sending the required information by the server to the client is termed as ‘response’. Figure 1 illustrates this Client-Server environment for web-based program.

To facilitate the development of software for different environment, Sun Microsystems announced a new architecture for java, with separate editions for different types of software packages developed using java technology. Java 2 platform, Standard Edition (Java SE, previously known as J2SE) is for desktop and workstation devices. Java 2 platform, Micro Edition (Java ME, previously known as J2ME) is for small devices like cell phones. Java 2 platform, Enterprise Edition (Java EE, previously known as J2EE) is for server based applications[6]. Java Server Page (JSP) and Java Servlet are some of the technologies included in Java EE and they are managed by web container. Web container is the software package that provides the runtime environment for executing JSP and servlets. Web container translates the request and response into standard Java objects. Java server page is the document that embeds some elements written in the java code in a HTML page (Java server page document has the extension ‘.jsp’). When the web container encounters JSP pages, it processes the java code, turns the JSP into a servlet and then executes that servlet. By this procedure, the web
Figure 2. Overview of the Java EE technologies for the development of software package for server.

container substitutes the process output for the embedded java code elements and converts the page into HTML document and sending that HTML document to client. Servlet is a java class inherited from a special java class known as 'HttpServlet'. Servlet can run only under web container. Servlet process the request from the client and sending the process results to the client in the form of response. JSP and servlets access database through JDBC. Figure 2 shows the high level view of all the technologies and their relationships.

The process of software development packages for scientific and commercial application is divided into a set of tiers as shown in figure 2. Java EE identifies three: the client tier, the middle tier and the Information system tier[6]. This logical separation, with well defined interfaces, makes it possible to build scalable applications. Initially one or more tiers can be running on the same physical server machine. With increased demands, the tiers can be separated and distributed over multiple servers without modifying the code, just by changing the configuration. The client tier contains browsers, regular GUI based Java programs and Java enabled smart devices such as cellular phone or PDA. A browser and Java programs use HTTP to communicate with web container. Smart devices use Wireless Access protocol (WAP) and that is converted into HTTP to communicate with the web container. The middle tier provides the client services through the web container. A client that communicates through HTTP with the server uses components in the web container, such as servlets and JSP pages, as entry points to the program modules. The Information System (IS) tier holds the business data. Typically, it consists of one or more relational database management servers. The middle tier uses Java EE APIs such as JDBC to interact with the IS tier.

2.1 The MVC Model

In addition to the separation of responsibilities into different tiers, Java EE also encourages the use of the Model-View-Controller (MVC) design model when designing the software packages for server-based programs. The key point of using MVC is to separate components into three different units: the Model, the View and the Controller. In server-based programs, we commonly classify the parts of the application as: business logic, presentation and request processing. Business logic is the term used for the manipulation and processing of the data. Presentation refers to how the information is displayed to the user. Request processing is what ties the business logic and presentation parts together. In MVC terms, the Model corresponds to business logic and data, the View to the
presentation logic and the Controller to the request processing. In most of the software packages, the data structure and logic (the Model) are typically the stable part, while presentation of the data (the View) changes often. The advantage of MVC model is due to the separation of Model and View, while modifying the View the Model part does not get disturbed. Another common example of why presentation should be separated from the business logic is that it may be required to present the data in different languages or present different subsets of the data to internal and external users. Access to the data through new types of devices, such as cell phones and Personal Digital Assistants (PDA), is the latest trend. Each client type requires its own presentation format. Separating business logic from presentation makes it easier to evolve the complete package of codes as the requirement changes. New presentation interfaces can be developed without touching the business logic.

Even though MVC model was originally developed for standalone GUI based Java programs, it translates fairly well into the multi tier application domain of Java EE. The user interacts with the Controller to ask for things to be done and the Controller relays these requests to the Model in a client-type independent way. The result of the request is then presented to the client. If the client happens to be a browser, it gets the result through HTTP response. If the Client happens to be a GUI based Java code, this code may include a View Component that communicates directly with the Model to get its new state and render it on the screen.

3. Database Generation using ‘Plppy’ Program.
‘Plppy’ is a GUI based program developed for MS-Windows to view the transport and thermodynamic properties of gases and gas mixtures used in thermal plasma processes. This program also helps to get the properties data either in tabular form or in the form of cubic polynomial coefficients in different temperature segments over the temperature range 300 – 30000K. The typical display of the main window of the ‘Plppy’ is shown in figure 3.

As can be seen from figure 3, we can select the required properties using the tool bar buttons provided in left side of the main window. We can select the required gas mixtures using the toolbar buttons provided at the top. We can view the required property data of many gas/gas mixtures simultaneously but we can view only one property data for the given gases/gas mixtures at a time.

Data is available for the following properties in ‘Plppy’ program.
1. Thermodynamic Properties
   Density, Specific Heat and Enthalpy
2. Transport Properties
   Viscosity, Thermal Conductivity and Electrical Conductivity

Data is provided for the following gas/gas mixtures in ‘Plppy’ program.
1. Argon
2. Nitrogen
3. Oxygen
4. Hydrogen
5. Helium
6. Air
7. Argon-Air( 25%, 50%, 75% of air by volume in the mixture)
8. Argon-Nitrogen(25%, 50%, 75% of air by volume in the mixture)
9. Argon-Oxygen(25%, 50%, 75% of air by volume in the mixture)
10. Argon-Hydrogen(25%, 50%, 75% of air by volume in the mixture)
11. Argon-Helium(25%, 50%, 75% of air by volume in the mixture)

The published properties data which is available in tabular form is obtained from the author [1-4], as a function of temperature over the range 300-30000K with step size of 100K. The temperature range is divided into many segments and in each segment the property data is fitted into
Figure 3. The main window of the GUI based program ‘Plppy’ showing the thermal conductivity of Ar-H2 gas mixtures over the temperature range from 300 K to 30000 K.

Figure 4. Saving the cubic polynomial coefficients for specific heat of Argon gas in a text file using ‘Plppy’ program.
cubic polynomial function using ‘CurveExpert’ software[5]. Hence we get a set of cubic polynomial coefficients over the range 300-30000K and the ‘Plppy’ program uses this polynomial coefficients to plot the property graph for the given gas/gas mixture. The program helps the user to save the set of cubic polynomial coefficients for the given gas in a text file using ‘File/Save as Coefficients’ menu command. For example, figure 4 shows how the cubic polynomial coefficients for the specific heat of Argon gas are stored in a text file.

Data file, which is in ASCII form, is generated to save the cubic polynomial coefficients for each property for the given gas/gas mixture and this data file generation is repeated for all the gases/gas mixtures. Using these data files, MS-Access database documents are created for all the gases/gas mixtures by a JAVA program using JDBC technique.
**Figure 7.** Form displayed initially to enter the user’s credential while attempting to get the transport and thermodynamic properties of High temperature gases/gas mixtures.

**Figure 8.** Form displayed to the user to enter parameters to get the transport and thermodynamic properties of High temperature gases/gas mixtures.
4. Design and the Development of program modules

The task of developing a package of the web based program modules to obtain the required property of high temperature gases/gas mixtures is achieved by developing the java codes for servlets, java server pages, java script documents and HTML documents[7]. The overview of the development strategy is illustrated in figure 5. As can be seen from the figure 5 the requirement of client is only executing a browser. All other requirement goes to the remote server, which can be connected to the client either through LAN or internet. In the server machine, it is required to install the following software and documents.

1. Java Virtual Machine (JVM) :- This is the run time environment for executing codes written in JAVA. This is provided by Sun Microsystems.
2. Apache Tomcat :- This is the web container for running servlets and JSP pages. This is provided by Apache Software Foundations group
3. Package of codes and documents specific to our requirements
4. MS-Access Database files.

The version of JVM should be compatible with the Apache Tomcat web container[8]. MS-Access database files are registered with ODBC Data Source Administrator. This is required for the JDBC 4 specifications. For the Windows-XP, this is done through control panel using the ODBC Data Source Administrator from the Administration tool group. All the data base files are registered as system data source and the driver for MS-Access database is selected while registering.

The entire working procedure for the program modules and the role of each module is illustrated in figure 6. Let us assume the web container, which is the Apache Tomcat server here, is activated. Initially user access the HTML document 'ThPlasma.html' stored in the server using the browser. This HTML document loads the java server page 'login.jsp' and the user login form is displayed in the client browser as shown in figure 7. Then the user credentials (username and password) are validated by a servlet ‘UserLogin’ and this servlet verifies the user credential using the ‘UserList’ database document. If the user’s validation gets succeeded, ‘UserLogin’ directs the control
Figure 10. Typical results obtained from the server in the form of GIF Image for the parameters shown in figure 8 

to another java server page ‘ThPlasma.jsp’ that displays the form in the client browser as shown in figure 8. As can be seen from fig. 8, user can select the gases/gas mixtures, property, temperature range and type of output. One of the type of output is in graphical form (GIF Image). The other types of output are in the tabular form of either actual property values as a function of temperature or cubic polynomial coefficients for different temperature segments. Form validation is done using java script, so that the user has properly selected the parameters. Once the validation is confirmed, the java server page directs the request to the main servlet ‘ThPlasma’ present in the archived package ‘ThPlasma.jar’ to get the property values. The flow chart illustrated in the figure 9 indicates the working procedure of the ‘ThPlasma’ servlet. ‘ThPlasma’ servlet access the database documents through JDBC to get the cubic polynomial coefficients to calculate the property values as a function of temperature. Figure 10 shows the graphs created by the ‘ThPlasma’ servlet and communicated to the user in the client browser. For this graphs the parameters selected by the user is shown in figure 8. It is to be noted here that the java server pages ‘login.jsp’, ‘ThPlasma.jsp’ and the servlet ‘UserLogin’ are acting as the Controller part of the MVC model described earlier. ‘ThPlasma’ servlet is acting as the Model part of MVC model.

5. Conclusion

The package ‘ThPlasma’ that consists of different modules like servelets, java server pages, java scripts and HTML documents is developed. This package helps the user to view the thermodynamic and transport properties of gases like Argon, Nitrogen, Oxygen, Hydrogen and helium and gas mixtures like Air, Argon-Air, Argon-Nitrogen, Argon-Oxygen, Argon-Hydrogen, Argon-Helium. This package helps to view the properties in the form of graphs, which is a function of temperature over the range from 300K to 30000K. This package also helps to obtain the property
values of the selected gases/gas mixtures in tabular form. The other alternative provided by this package is getting the cubic polynomial coefficients for the selected property and the selected gas/gas mixture in different temperature segments in the temperature range mentioned before. The developed package is tested with Apache Tomcat web container.

Reference

[1] A. B. Murphy and C. J. Arundell, Transport coefficients of argon, nitrogen, oxygen, argon-nitrogen and argon-oxygen plasmas, *Plasma Chemistry and Plasma Processing*. 14 451-490, 1994.
[2] A. B. Murphy, Transport coefficients of air, argon-air, nitrogen-air, and oxygen-air plasmas, *Plasma Chemistry and Plasma Processing*. 15 279-307, 1995.
[3] A. B. Murphy, Transport coefficients of helium and argon-helium plasmas, *IEEE Transactions on Plasma Science*. 25 809-814 (1997).
[4] A. B. Murphy, Transport coefficients of hydrogen and argon-hydrogen mixtures, *Plasma Chemistry and Plasma Processing*. 20 279-297 (2000).
[5] Web Page :- http://home.comcast.net/~curveexpert/. For getting information about CurveExpert 1.3 which is a comprehensive curve fitting system for MS-Windows
[6] Hans Bergsten, 2004 *Java Server Pages* (Mumbai: Shroff Publishers & Distributors Ltd)
[7] Suresh Rajagopalan, Ramesh Rajamani, Ramesh Krishnaswamy and Sridhar Vijendran 2002 *Java Servlet Programming Bible*, (NewDelhi: John Wiley & Sons)
[8] Vivek Chopra, Singh Li and Jeff Genender 2007 *Professional Apache Tomcat 6* (NewDelhi: Wiley India Pvt. Ltd)