Fusion Engineering and Design xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

Use of EPICS and Python technology for the development of a computational toolkit for high heat flux testing of plasma facing components

Ritesh Sugandhi*, Rajamannar Swamy, Samir Khirwadkar

Institute for Plasma Research, Gandhinagar 382428, India

HIGHLIGHTS

- An integrated approach to software development for computational processing and experimental control.
- Use of open source, cross platform, robust and advanced tools for computational code development.
- Prediction of optimized process parameters for critical heat flux model.
- Virtual experimentation for high heat flux testing of plasma facing components.

ARTICLE INFO

Article history:
Received 19 June 2015
Received in revised form 15 February 2016
Accepted 21 April 2016
Available online xxx

Keywords:
Critical heat flux
Plasma facing component
Numerical optimization
EPICS
Python

ABSTRACT

The high heat flux testing and characterization of the divertor and first wall components are a challenging engineering problem of a tokamak. These components are subject to steady state and transient heat load of high magnitude. Therefore, the accurate prediction and control of the cooling parameters is crucial to prevent burnout. The prediction of the cooling parameters is based on the numerical solution of the critical heat flux (CHF) model. In a test facility for high heat flux testing of plasma facing components (PFC), the integration of computations and experimental control is an essential requirement. Experimental physics and industrial control system (EPICS) provides powerful tools for steering controls, data simulation, hardware interfacing and wider usability. Python provides an open source alternative for numerical computations and scripting. We have integrated these two open source technologies to develop a graphical software for a typical high heat flux experiment. The implementation uses EPICS based tools namely IOC (I/O controller) server, control system studio (CSS) and Python based tools namely Numpy, Scipy, Matplotlib and NOSE. EPICS and Python are integrated using PyEpics library. This toolkit is currently under operation at high heat flux test facility at Institute for Plasma Research (IPR) and is also useful for the experimental labs working in the similar research areas. The paper reports the software architectural design, implementation tools and rationale for their selection, test and validation.

1. Introduction

Divertor system of a tokamak consists of systematically arranged plasma facing mockups, which is utilized to exhaust helium ash and excessive heat absorption to improve the plasma performance. The divertor design is complex due to the high heat load (~5–10 MW/m^2), withstanding material properties of the mockup and the efficiency of the heat transfer of the cooling loop. A mockup is made up of carbon fiber-reinforced carbon composite (CFC) or tungsten alloy joined with actively cooled copper alloy as heat sink material. The performance of the mockup is characterized by thermal–hydraulic response for the mockup under steady state and transient heat load. The cooling channel of the mockup can have many configurations such as bare channel, swirl tube and hyper-vapotrons. Worldwide research facilities are established for high heat flux testing namely JUDITH-II and EB-1200. Recently, high heat flux test facility (HHFTF) [1–3] has been commissioned at IPR. The integration of computations, simulations and experimental control is an essential requirement of the facility. Numerical computations facilitate testing of all the feasible scenarios and are useful for prediction of the experiments, where operational cost

* Corresponding author.
E-mail addresses: ritesh@ipr.res.in (R. Sugandhi), rajamannar@ipr.res.in (R. Swamy), sameer@ipr.res.in (S. Khirwadkar).

http://dx.doi.org/10.1016/j.fusengdes.2016.04.036
0920-3796/

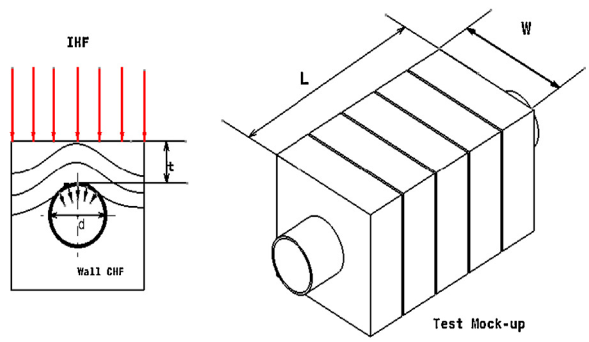

Fig. 1. Heat load in the smooth tube.

is very high. Recent advancements in the open source technologies provide a robust solution for the system design. EPICS and Python are such emerging technologies of choice in fusion world. The work on developing an integrated software for prediction of cooling parameters and simulation of the experimental testing of a mockup, is undertaken and reported in this paper.

The paper is organized as follows: Section 2 describes the computational model of the heat transfer followed by the design of the toolkit in Section 3. The results are discussed in Section 4 followed by future scope of work in Section 5 and conclusion in Section 6. All symbols are listed in Appendix A.

2. Computational model

CHF describes the loss of liquid layer or phase change of the coolant at the wall of the cooling channel, which leads to decrease in the efficiency of the heat transfer causing burn-out. Heat transfer is maximized at the CHF point and is drastically degraded afterwards. The accurate prediction of CHF is must for the safe design of the plasma facing components [4,5]. Heat transfer model is explained by Nukiyama curve [6], which defines pre-CHF, CHF and post-CHF regions. The computational model of the CHF is defined by Tong-75 correlation [7]. It is selected because (a) it satisfactorily incorporate the associated thermal and hydro-dynamic effects (b) predicts the local CHF (c) has been recommended for fusion CHF predictions and (d) compares well with the data. The equation of CHF (CHF_w) is given by:

$$CHF_w = 0.23 \times f \times G \times H_{fg} \times \left(1 + 0.00216 \times \left(\frac{P}{P_c}\right)^{1.8} \times Re^{0.5} \times Ja\right) \times C_f \quad (1)$$

where dimensionless coefficients Reynolds number (Re), Jakob number (Ja), Prandtl number (Pr) and Friction factor (f) are given by:

$$Re = \frac{G \times d}{\mu_f}, \quad Ja = \frac{\rho_f}{\rho_g} \times \frac{C_p \times (T_{sat} - T)}{H_{fg}},$$

$$f = 8 \times Re^{-0.6} \times \left(\frac{d_h}{d_0}\right)^{0.32}, \quad Pr = \frac{C_p \times \mu_f}{k}$$

Incident heat flux (IHF) is defined at the surface of the exposed mock up. Since, in PFC, only one side is heated by incident flux, it is assumed that the heat load at the tube wall corresponds to the area proportion. This assumption is valid in case of double side loading. The heat distribution in a mono block is described in Fig. 1. Inner wall heat flux ($IWHF$) is given by:

$$IWHF = \frac{IHF \times A_{surface}}{A_{tube}} \quad (2)$$

where

$$IHF = \frac{Power}{A_{surface}}, \quad A_{surface} = W \times L, \quad A_{tube} = \pi \times \frac{d}{2} \times L \quad (3)$$

The CHF_{reqd} defines a value for which CHF needs to be optimized. The system should be operated such that the safety margin between CHF_w and $IWHF$ to be maintained. CHF_{reqd} is given by Eq. (4) and optimization problem is given by Eq. (5). The optimization is subject to real world constraints of the experimental system on pressure (P), temperature (T), flow (F) and diameter of the tube (d). The optimization problem is solved using constrained optimization by linear approximation (COBYLA) [8] technique.

$$CHF_{reqd} = IWHF \times F_p \quad (4)$$

$$\underset{P,T,F,d}{\text{Maximize}} \quad CHF_{reqd}(P, T, F, d) \quad (5)$$

The heat transfer coefficient (h) is calculated using sider-tate correlation and is given by:

$$h = \frac{k}{D_h} \times 0.0027 \times Re^{0.8} \times Pr^{0.33} \times \frac{\mu_b}{\mu_w}^{0.14} \quad (6)$$

Absorbed heat load (Q_{abs}) is given by:

$$Q_{abs} = h \times A_{tube} \times (T_s - T_b), \quad Tb = \frac{T_{in} + T_{out}}{2} \quad (7)$$

The CHF calculation are done by using data tables representing the thermo-physical properties of coolant (water in our case), taken from the database of National Institute of Standards and Technologies (NIST) [9]. The fluid data tables represent the experimental relationship as follows:

(a) Temperature (T) vs. specific heat (C_p), density (ρ), thermal conductivity (k).
(b) Pressure (P) vs. saturated temperature (T_{sat}), latent heat of vaporization (H_{fg}).
(c) Saturated temperature (T_{sat}) vs. dynamic viscosity (μ_f).

3. Software description

3.1. Requirements

Following are the requirements for the design:

(a) Implement the virtual experiment life cycle consists of compute – simulate – measure – analysis.
(b) Automate the curve fitting process for the specified goodness of fit and abstract the complex inter-parameters dependencies described in Section 2.
(c) Provide software routines for heat transfer computations, constraint optimization on various parameters, data plotting and cross validation for CHF_w.
(d) Provide the virtual experimental execution on the obtained set points, cooling loop configuration, archival and analysis of the simulated data.
(e) Provide robust framework for easy interface to the I&C sensors and actuators.

3.2. Design

Fig. 2 represents the software block diagram outlining the software modules and their dependence. Fig. 3 represents the activity flow diagram representing the virtual experimental cycle carried out before experiments at HHFTF.