Energy 94 (2016) 380–390

Contents lists available at ScienceDirect

Energy

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

A new modular procedure for industrial plant simulations and its reliable implementation

C. Carcasci [a], L. Marini [a], B. Morini [a], M. Porcelli [b, *]

[a] Università degli Studi di Firenze, Dipartimento di Ingegneria Industriale, via di S. Marta 3 e Viale Morgagni 40, 50134 Firenze, Italy
[b] Università di Bologna, Dipartimento di Matematica, Piazza di Porta S. Donato 5, 40127 Bologna, Italy

ARTICLE INFO

Article history:
Received 29 April 2015
Received in revised form
23 September 2015
Accepted 24 October 2015
Available online 28 November 2015

Keywords:
Industrial plants simulation
Modular 0/1-D codes
Fully implicit methods
Constrained nonlinear systems of equations
Trust-region Gauss–Newton methods

ABSTRACT

Modeling of industrial plants, and especially energy systems, has become increasingly important in industrial engineering and the need for accurate information on their behavior has grown along with the complexity of the industrial processes. Consequently, accurate and flexible simulation tools became essential yielding the development of modular codes. The aim of this work is to propose a new modular mathematical modeling for industrial plant simulation and its reliable numerical implementation. Regardless of their layout, a large class of plant's configurations is modeled by a library of elementary parts; then the physical properties, compositions of the working fluid, and plant's performance are estimated. Each plant component is represented by equations modeling fundamental mechanical and thermodynamic laws and giving rise to a system of algebraic nonlinear equations; remarkably, suitable restrictions on the variables of such nonlinear equations are imposed to guarantee solutions of physical meaning. The proposed numerical procedure combines an outer iterative process which refines plants characteristic parameters and an inner one which solves the arising nonlinear systems and consists of a trust-region solver for bound-constrained nonlinear equalities. The new procedure has been validated performing simulations against an existing modular tool on two compression train arrangements with both series and parallel-mounted compressors.

1. Introduction

Industrial plants are subject to standard requirements as low equipment costs, high energy conversion/transmission efficiency, low environmental impact and high operational flexibility. In particular, in industrial plant design and in-service behavior prediction, high calculation accuracy and competitive computational time are fundamental to meet the customers' needs. These are the reasons why traditional methods for such simulations involve the use of numerical 0/1-D codes, known to fully satisfy the above requirements. In particular, the dedicated approach leads to procedures for specific plant configurations where either none or a few input data are allowed to vary. Over the last decades, this approach has progressively been replaced by a modular one that can handle general plant's layout and general input data.

Recently, there has been a renewed interest in global plant landscape driven by the increasing demand for low-cost energy and reduced environmental impact; this has led to continuous efforts for enhancing the elementary plants' components and to the theoretical and practical study of alternative thermodynamic cycles. On one side, this process resulted in a general complication of the plants' arrangements and, correspondingly, in a high demand of flexible tools to numerically estimate the plants' performance. On the other hand, it yielded to the description of a large variety of plant solutions as a combination of a finite number of elementary components (pumps, heat exchangers, valves, turbines, compressors) connected with each other. Therefore a numerical code equipped with a database of elements representing their physical behavior and suited to combine these with general criteria, turns out to be much more versatile than a code designed for a specific plant's configuration. These types of codes are known as *modular* codes and are generally characterized by the following properties [1]. They must be able to:

- create a plant configuration without requiring a new program source;
- handle any combination of input data if a sufficient number of parameters for the plant's solution is provided;
- find the characteristic parameters of the elementary components, even with an increased number of input data.

* Corresponding author.
E-mail addresses: carlo.carcasci@unifi.it (C. Carcasci), leopoldo.marini@unifi.it (L. Marini), benedetta.morini@unifi.it (B. Morini), margherita.porcelli@unibo.it (M. Porcelli).

http://dx.doi.org/10.1016/j.energy.2015.10.122
0360-5442/

Since the early 1990s many modular codes for plants' simulation have been developed to fulfill the growing needs of flexible and fast numerical tools for the study of complex flow networks; firstly Perz [2] built an elemental code for mass and heat balances, named IPSEpro, that has been subsequently developed with commercial purposes by Simtech Simulation Technology [3]. The program, due to its simplifications, is able to model only the behavior of ideal gas [4]. Carcasci and Facchini [1] proposed a modular code for applied research that, grown over the years, handles thermodynamic [5], design and off-design analysis of industrial plants [6,7]. Falcetta and Sciubba [8,9], similarly, developed a modular tool, named CAMEL, that has become a commercial code owned by Altran [10]. Also Carapellucci and Cau studied a modular procedure based on fundamental thermodynamic relations, including real gas behavior [11], mainly used for power plants simulations [12]. Many other modular-based codes have been developed for commercial uses since their origin; examples include THERMOFLEX, property of Thermoflow Inc. [13], that is a fully flexible program for heat balance modeling and engineering, particularly suited to model both conventional [14] and unconventional power plants, like solar ones [15]; GE's GateCycle [16], a professional tool for both the gas and steam sides of power plant design and analysis [17]; Prosim, a modular simulation and design environment for power processes [18], developed by Endat Oy (Prosim, www.endat.fi). The great advantages of the modular approach have yielded the development of a whole class of object-oriented programming languages, among which the most notably is Modelica, introduced in 1996 within the project ESPRIT [19]. Modelica deals with component-oriented modeling of complex physical systems consisting of mechanical, electrical, hydraulic, thermal and control equipment [20]. The algorithms and programming approaches adopted in both such commercial and applied research-based codes are rather obscure and often confidential; this makes extremely hard to gain an in-depth understanding of their properties and potentials in plants' simulations.

One of the issues which greatly influences the performance of the codes but has not been thoroughly discussed in the literature consists in the numerical solution of the equations for process simulation. In particular, a common task of all the above-mentioned codes is the solution of a set of equations, including differential and algebraic equations, that represents the physical behavior of the modeled problem. Such set can be either split into subsets of equations, each coming from a particular module [21], that are solved using a proper sequential approach (alike the approach adopted in dedicated codes) or solved simultaneously by a *parallel/full implicit mode*. None of above papers offer details for this algorithmic phase.

Differential equations have a marginal role in industrial plants' simulations, since the modular codes are generally zero dimensional and model steady state flows, whereby neither spatial nor temporal evolution of the phenomena is taken into account within each element. On the contrary, systems of nonlinear equations (i.e., systems where at least one equation is not linear) constitute the mathematical models for a surprisingly large number of problems of real concern, as they model both the behavior of dynamic and thermodynamic systems, through the discretization of ordinary or partial differential equations, and equilibrium systems, see e.g. Refs. [1,2,4,10,18,22,23].

Newton method and its variants are arguably the most popular class of procedures for solving nonlinear systems of algebraic equations [24] and they are used in most of the above codes [4,10,18,22,23]. It is well-known that Newton method is an iterative procedure and its convergence depends critically on the vicinity of the initial guess to a solution of the nonlinear system. For practical applications in modular codes, this feature may represent a severe limitation as the nonlinearity of the equations and the number of unknowns make the location of the roots highly difficult. In order to enhance convergence, Newton's method is combined with so-called globalization strategies which include linesearch and trust-regions methods, see e.g. Refs. [24–31]. An alternative solution strategy, implemented in codes for industrial plants' simulations [1,2], consists in a simplification of the mathematical problem to be solved where, by means of first-order Taylor expansion, nonlinear equations are replaced with linear equations. The resulting problem is a linear system which can be solved with standard algorithms such as Gauss-Jordan Elimination or Lower-Upper Decomposition [1,2]. Clearly, though computational cheaper than Newton method, this approach may provide an inaccurate approximation to a solution of the original problem and does not overcome the need to locate solutions of the nonlinear system.

A further issue that requires modification of standard procedures for solving nonlinear systems derives from the fact the systems of our interest are constrained. Specifically, bounds have to be imposed to find a solution of physical meaning, e.g., a solution where absolute pressures are positive, and possibly to restrict the search space for a desired solution, see e.g. Refs. [26,32].

The aim of this paper is to provide a detailed overview of a new modular procedure for industrial plants' simulation that can handle a broad class of plant's layouts through a wide library of elementary components, and determine the physical properties and composition of the working fluid, as well as plant's performance, in steady state operational conditions. Each module of the developed code has been made as independent as possible from the others, enhancing flexibility and allowing for upgrades; e.g. the solver of the arising nonlinear systems of equations can be readily replaced or improved without altering the other parts of the code. The core of this implementation is given by use of the nonlinear optimization solver TRESNEI for bound constrained nonlinear least-squares problems [29]. This solver implements a trust-region Gauss–Newton method and is suitable for the solution of "smooth" problems, that is problems described by continuously differentiable functions. It provides enhancements with respect to standard solvers for nonlinear systems in the following respects:

- being a solver for bound constrained problems, once proper upper and lower bounds for the variables are fixed, it prevents the computation of undesirable solutions lying outside from the feasible solution's domain;
- by implementing a globally convergent method, it avoids the tricky issue of selecting an initial guess close enough to the problem's solution.

Moreover, since our approach does not rely on simplified versions of the nonlinear system, the solutions computed are expect to be more accurate than those obtained with the approach in Refs. [1,2].

In this paper, a thorough description of both our modular procedure and TRESNEI's algorithm is provided. Our goal is to offer a scheme that can serve as a template to users interested in reproducing, and possibly adapting, our code. The performance of the proposed modular procedure is illustrated on two compression train arrangements with both series and parallel-mounted compressors; the results obtained have been compared with ESMS [1], a pre-existing in-house modular tool, based on a Gauss-Jordan solver, that has been widely validated over a broad range of industrial plants' problems [5–7]. The comparison shows a good agreement between the results of the two codes and comparable computational speed.