Dynamics of Continuous, Discrete and Impulsive Systems
Series **B:** *Applications & Algorithms* **18 (2011)** 365-381
 http://www.watam.org

A NUMERICAL ALGORITHM FOR FLAME SIMULATION

Piotr Opioła

Institute of Computer Science
Jagiellonian University
ul. Łojasiewicza 6, 30-348 Kraków, Poland
opiola@ii.uj.edu.pl

Abstract. This work presents a numerical algorithm for flame simulation. A compressible fluid model of a laminar flame is developed from the Navier-Stokes equations along with energy relations and reaction rates. The algorithm is based on a finite difference scheme. In order to avoid the stability condition due to convection, a forward semi-Lagrangian approach is implemented. Numerical results for a premixed, methane-oxygen flame are presented.

Keywords. Computational fluid dynamics; Flame simulation.

AMS (MOS) subject classification: 35Q35; 76M20; 76M12; 76N15.

1 Introduction

Flame simulation has many applications, such as fire protection systems or engine design. However, precise simulation of a flame is difficult because of significant changes of energy during the combustion reaction, changes of chemical species, and turbulence. Standard computational fluid dynamics techniques have to be modified to satisfy specific flame simulation requirements. In this work, we present a general method for laminar flame simulation. While the turbulence often plays an important role in a combustion process, there are still many cases in which the flame is laminar, e.g. the Bunsen burner [3].

Fluid flow models usually assume constant density, which is inappropriate for a flame model. The density of gases inside a flame can vary significantly because of the changes in temperature. Variable density of the modelled fluid makes the standard pressure-correction methods invalid in this case. Introducing compressibility in the model gives rise to new phenomena such as the presence of acoustic waves, which leads to significant time step-size restriction. There are implicit methods for avoiding this restriction (see [8], [12]). We propose a much simpler approach, which does not offer high precision, but is relatively easy to implement and efficient. Instead of limiting the time step, we limit the change of the velocity vector.