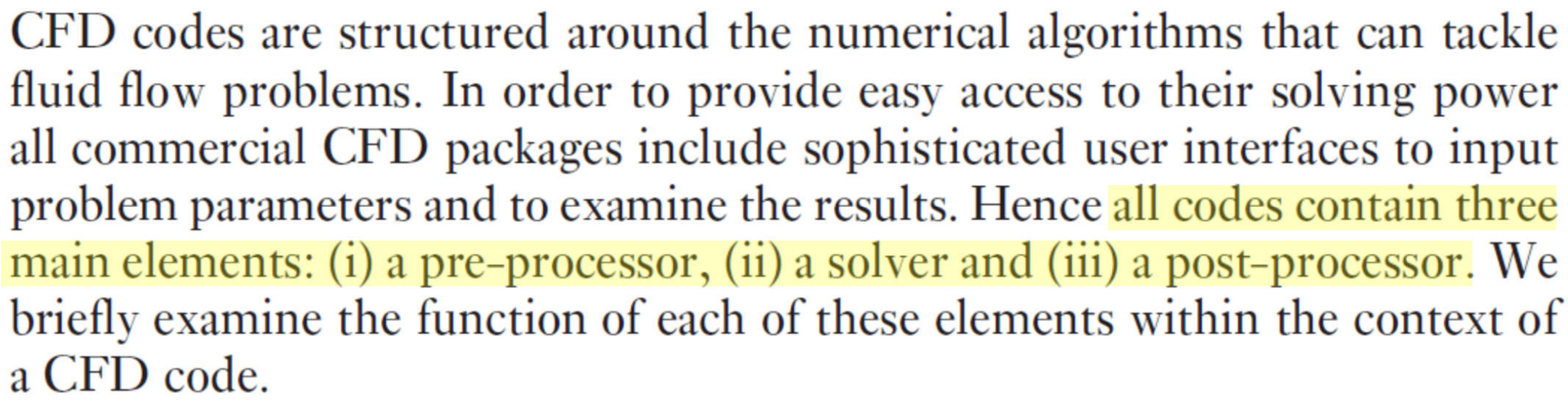
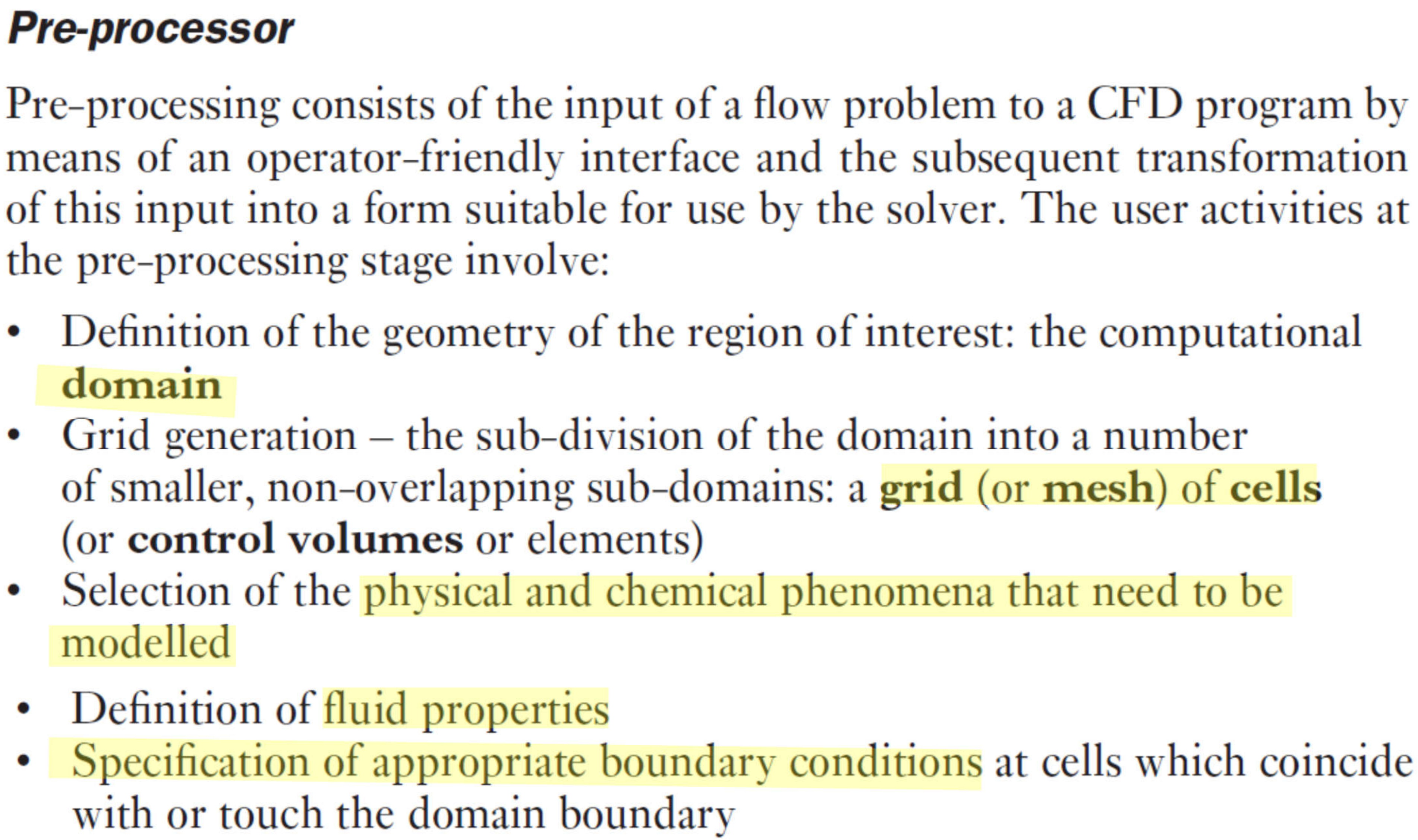
1.2 How does a CFD code work?



CFD codes are structured around the numerical algorithms that can tackle fluid flow problems. In order to provide easy access to their solving power all commercial CFD packages include sophisticated user interfaces to input problem parameters and to examine the results. Hence all codes contain three main elements: (i) a pre-processor, (ii) a solver and (iii) a post-processor. We briefly examine the function of each of these elements within the context of a CFD code.

1.2 How does a CFD code work?

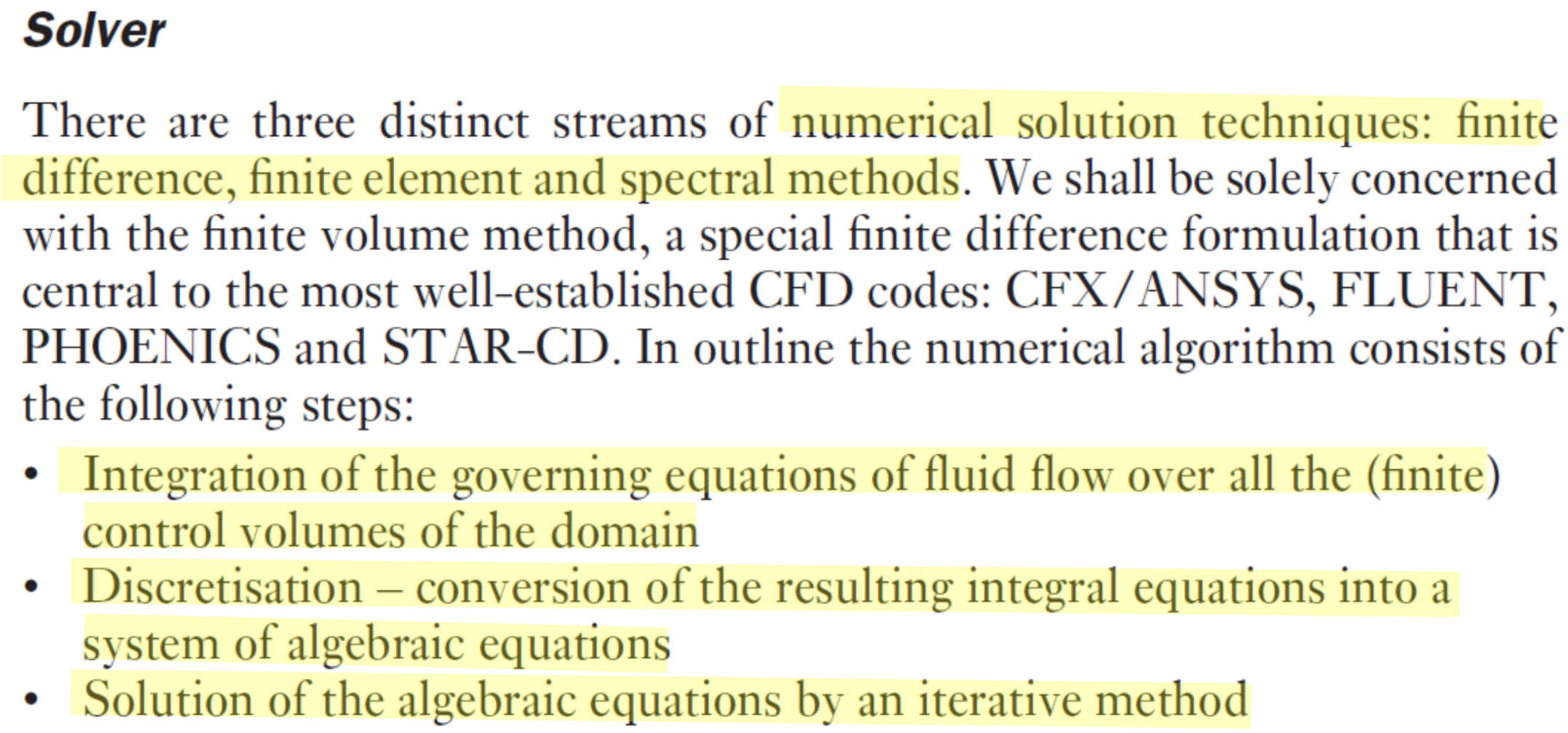


Pre-processor

Pre-processing consists of the input of a flow problem to a CFD program by means of an operator-friendly interface and the subsequent transformation of this input into a form suitable for use by the solver. The user activities at the pre-processing stage involve:

- Definition of the geometry of the region of interest: the computational **domain**
- Grid generation – the sub-division of the domain into a number of smaller, non-overlapping sub-domains: a **grid** (or **mesh**) of **cells** (or **control volumes** or elements)
- Selection of the physical and chemical phenomena that need to be modelled
- Definition of fluid properties
- Specification of appropriate boundary conditions at cells which coincide with or touch the domain boundary

1.2 How does a CFD code work?



Solver

There are three distinct streams of numerical solution techniques: finite difference, finite element and spectral methods. We shall be solely concerned with the finite volume method, a special finite difference formulation that is central to the most well-established CFD codes: CFX/ANSYS, FLUENT, PHOENICS and STAR-CD. In outline the numerical algorithm consists of the following steps:

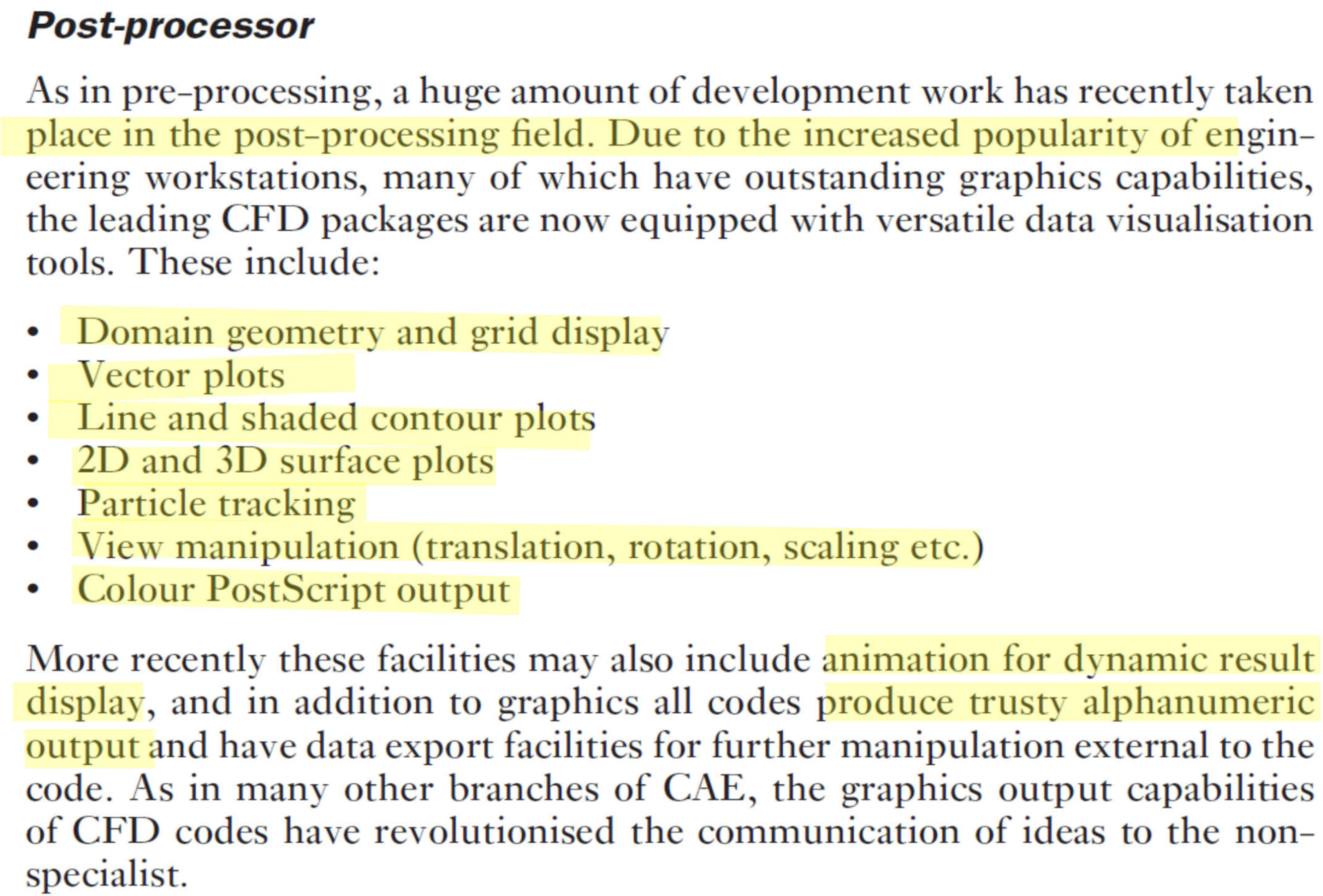
- Integration of the governing equations of fluid flow over all the (finite) control volumes of the domain
- Discretisation – conversion of the resulting integral equations into a system of algebraic equations
- Solution of the algebraic equations by an iterative method

1.2 How does a CFD code work?

The first step, the control volume integration, distinguishes the finite volume method from all other CFD techniques. The resulting statements express the (exact) conservation of relevant properties for each finite size cell. This clear relationship between the numerical algorithm and the underlying physical conservation principle forms one of the main attractions of the finite volume method and makes its concepts much more simple to understand by engineers than the finite element and spectral methods. The conservation of a general flow variable ϕ, e.g. a velocity component or enthalpy, within a finite control volume can be expressed as a balance between the various processes tending to increase or decrease it. In words we have:

$$\begin{bmatrix}\text{Rate of change}\\ \text{of } \phi \text{ in the}\\ \text{control volume}\\ \text{with respect to}\\ \text{time}\end{bmatrix} = \begin{bmatrix}\text{Net rate of}\\ \text{increase of}\\ \phi \text{ due to}\\ \text{convection into}\\ \text{the control}\\ \text{volume}\end{bmatrix} + \begin{bmatrix}\text{Net rate of}\\ \text{increase of}\\ \phi \text{ due to}\\ \text{diffusion into}\\ \text{the control}\\ \text{volume}\end{bmatrix} + \begin{bmatrix}\text{Net rate of}\\ \text{creation of}\\ \phi \text{ inside the}\\ \text{control}\\ \text{volume}\end{bmatrix}$$

1.2 How does a CFD code work?



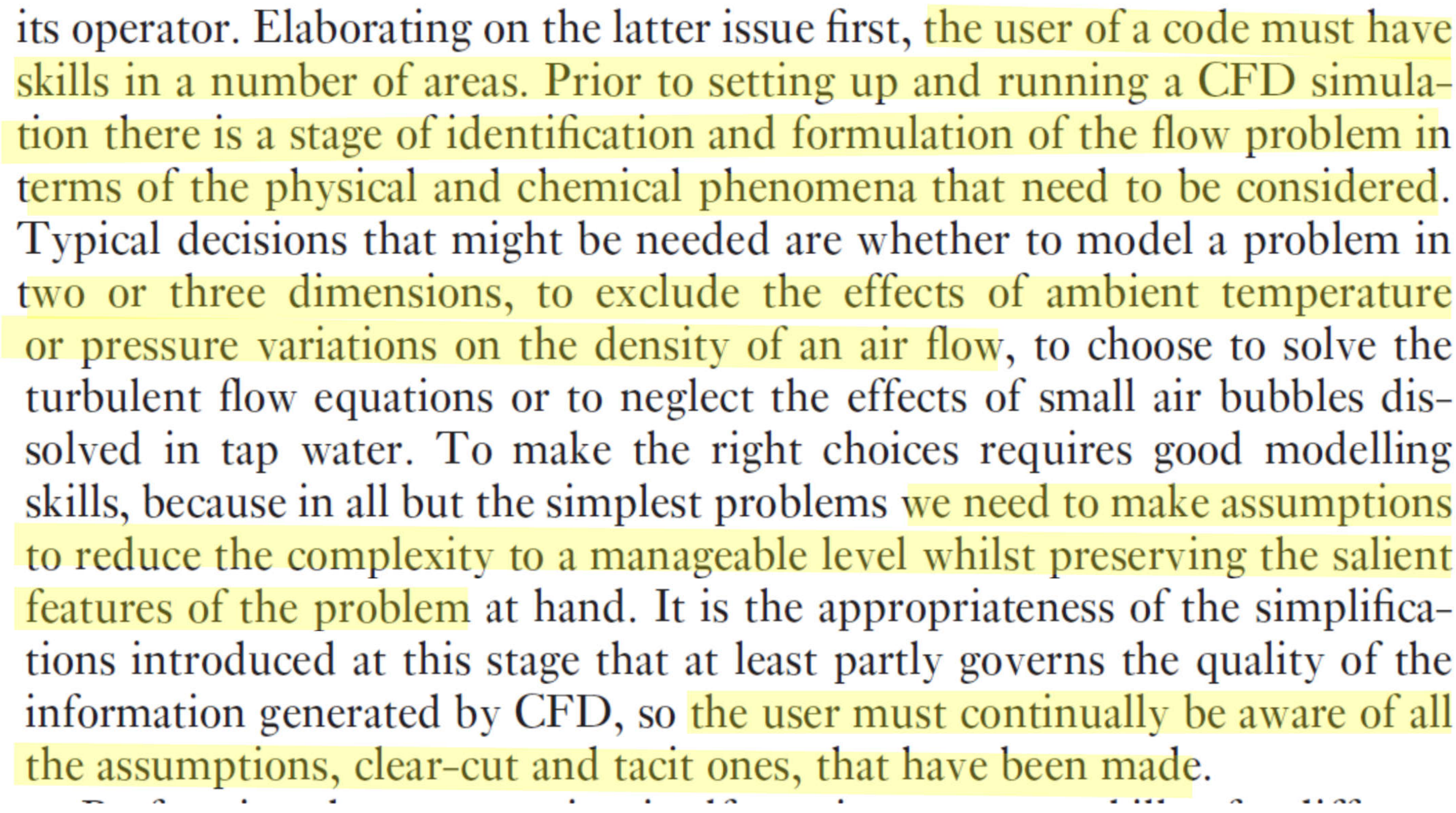
Post-processor

As in pre-processing, a huge amount of development work has recently taken place in the post-processing field. Due to the increased popularity of engineering workstations, many of which have outstanding graphics capabilities, the leading CFD packages are now equipped with versatile data visualisation tools. These include:

- Domain geometry and grid display
- Vector plots
- Line and shaded contour plots
- 2D and 3D surface plots
- Particle tracking
- View manipulation (translation, rotation, scaling etc.)
- Colour PostScript output

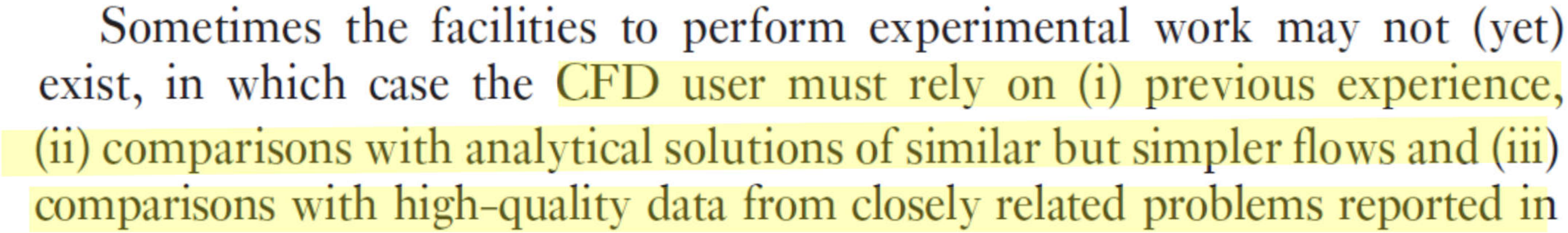
More recently these facilities may also include animation for dynamic result display, and in addition to graphics all codes produce trusty alphanumeric output and have data export facilities for further manipulation external to the code. As in many other branches of CAE, the graphics output capabilities of CFD codes have revolutionised the communication of ideas to the non-specialist.

1.3 Problem solving with CFD



its operator. Elaborating on the latter issue first, the user of a code must have skills in a number of areas. Prior to setting up and running a CFD simulation there is a stage of identification and formulation of the flow problem in terms of the physical and chemical phenomena that need to be considered. Typical decisions that might be needed are whether to model a problem in two or three dimensions, to exclude the effects of ambient temperature or pressure variations on the density of an air flow, to choose to solve the turbulent flow equations or to neglect the effects of small air bubbles dissolved in tap water. To make the right choices requires good modelling skills, because in all but the simplest problems we need to make assumptions to reduce the complexity to a manageable level whilst preserving the salient features of the problem at hand. It is the appropriateness of the simplifications introduced at this stage that at least partly governs the quality of the information generated by CFD, so the user must continually be aware of all the assumptions, clear-cut and tacit ones, that have been made.

1.3 Problem solving with CFD



Sometimes the facilities to perform experimental work may not (yet) exist, in which case the CFD user must rely on (i) previous experience, (ii) comparisons with analytical solutions of similar but simpler flows and (iii) comparisons with high-quality data from closely related problems reported in