SPE Brazil Flow Assurance Technology Congress
Rio de Janeiro, Brazil, November06 – 08 2024

Two-Phase Flow of Carbon-Dioxide in Pipelines: A GPU Accelerated Always Hyperbolic Two-Phase Flow Solver with Phase Change and Complex EOS

Farhad Nikfarjam[1], Gustavo E. O. Celis[2], Hamidreza Anbarlooei[1*], Daniel O. A. Cruz[2], Raiza G. Souza[3], Jorge Trujillo[3]
[1]Department of Applied Mathematics - Institute of Mathematics, Federal University of Rio de Janeiro, [*]hamidreza@im.ufrj.br
[2]Mechanical Engineering Program (COPPE), Federal University of Rio de Janeiro
[3]Upstream-Production Engineering, Galp Energy S.A., Lisbon, Portugal

Abstract
This paper introduces a numerical method for simulating two-phase compressible flow of CO_2, incorporating complex empirical equation of state (EOS) which is essential for understanding industrial processes like carbon capture, storage, and utilization (CCSU). To ensure robustness across various transient scenarios, particularly in accurately predicting phase changes during CO_2 transportation, we have developed a slug capturing method based on a consistently well-posed two-fluid 7-equation formulation. The Span-Wagner (SW) EOS is used to accurately express the complex behavior of CO_2, specially near the saturation line and critical point. However, the computational expense of this EOS necessitates exploring the look-up table method for efficient implementation. Due to conservative nature of the governing equations employed in the present work, it is also necessary to transfer this EOS from density-temperature space into the density-internal energy space. However, this transformation results in different sizes (order of magnitude) for the liquid, vapor, and two-phase regions, complicating the formation of a single table for the entire domain. To overcome this challenge, we have produced three separate tables (for liquid, vapor, and two-phase regions), ensuring consistent results across all three regions (with errors less than 1%) and avoiding the time-consuming iterative process typically required in two-phase regions. Furthermore, we have implemented the slug capturing scheme and table look-up method on the GPU framework, achieving up to two orders of magnitude speedup compared to CPU implementation. Benchmark simulations confirm the accuracy and efficiency of the tabulated EOS compared to other methods. This research effectively bridges the gap between accuracy and computational efficiency in simulating CO_2 behavior, crucial for optimizing CCSU strategies.

Keywords
CO_2 pipeline transport; Look-up table EOS; GPU-accelerated computing;

Introduction
Numerical simulation of two-phase compressible flow of CO_2 plays a crucial role in understanding the behavior of carbon dioxide in various industrial processes, particularly in carbon capture, storage and utilization (CCSU) initiatives aimed at mitigating greenhouse gas emissions. In the CCSU industry, to optimize the economic feasibility of storing and conveying CO_2, it is common to transport CO_2 either in a dense liquid state or in the supercritical state at high pressures [1,2]. However, pertinent processes like heat exchange with the environment or incidents such as pipeline rupture could lead to rapid transient effects, such as phase change. Therefore, comprehending the behavior of CO_2 within pipeline flows is crucial. For this purpose, we developed a new slug capturing method based on two-fluid 7-equation formulation. The solver is always hyperbolic, ensuring well-posedness across diverse scenarios. Moreover, our solver accurately predicts phase transitions, a crucial aspect in modeling CO_2 behavior during transport.
It's common to model CO_2 behavior using simplified equations of state (EOS) like Stiffened Gas (SG) or Nobel-Abel Stiffened Gas (NASG), along with Gibbs free energy to determine two-phase equilibrium. It is well-known that such simplified equations of state cannot accurately model the complex phase behavior of CO_2 and providing accurate CO_2 properties are crucial for a reliable simulation [3,4], especially near the critical points. The Span-Wagner (SW) EOS is well-known as the international reference for CO_2, covering a vast range of temperatures and pressures from triple-point temperature to 1100 K and pressures up to 800 MPa [5]. Using this EOS, it is feasible to accurately model CO_2 behavior at higher pressures and around critical points, which is crucial for CCSU applications.

However, while undeniably accurate, it is also computationally very expensive, especially for explicit time marching methods [6]. The look-up table methods are a good choice to address this issue. These methods comprise both accuracy and computational efficiency. The look-up tables provide pre-calculated thermodynamic properties of CO_2 over a wide range of pressures and temperatures, eliminating the need for complex and time-consuming calculations of different thermodynamics properties during the simulation cycles. This procedure accelerates the simulation process significantly, enabling researchers and engineers to explore various scenarios and optimize CCSU strategies efficiently. The current endeavor focuses on implementing a look-up table for our in-house developed 7-equation solver. Given that the two-fluid 7-equation solver is a complex explicit transient solver, it demands significant computational resources. Thus, all methods developed during present work have been implemented in GPU framework. The GPU-accelerated implementation achieves two orders of magnitude speedups which enables the approach to be employed in practical applications with pipelines spanning tens of kilometers [7,8].

The rest of this paper is organized as follows. First of all, the governing equations and numerical solution are outlined. Then, we discussed the look-up table method details, employed to calculate the thermodynamic properties of CO_2. A section is also dedicated to the implementation of GPU into the solver. Afterward, the numerical results of various problems are studied numerically and compared with numerical data available in literature. Finally, some concluding remarks are provided in last section.

Governing equations

The solver developed here is based on two-fluid 7-equation slug capturing methods. Slug capturing methods solves the two-fluid model equations in conjunction with a minimal set of empirical closure equations which can predict stratified, slug and transitional regimes [9]. They can be used to simulate transient phenomena such as start-up or shut down processes and are able to handle terrain and sever slugging naturally. It is common to simulate the intermittent two-phase flow using a two-fluid model, where the conservation of mass, momentum and energy, along with evolution of volume fraction will be solved for each phase as:

$$\begin{aligned}
&\frac{\partial \alpha_g}{\partial t} + u_i \frac{\partial \alpha_{\mathrm{g}}}{\partial \mathrm{x}} = \mathrm{r_p}(\mathrm{p_g} - \mathrm{p_l}),\\
&\frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}})}{\partial \mathrm{t}} + \frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}}\mathrm{u_k})}{\partial \mathrm{x}} = (-1)^{k+1}\dot{\mathrm{m}},\\
&\frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}}\mathrm{u_k})}{\partial \mathrm{t}} + \frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}}\mathrm{u_k^2} + \alpha_{\mathrm{k}}\mathrm{p_k})}{\partial \mathrm{x}} - p_{ik}\frac{\partial \alpha_k}{\partial x} = \mathrm{S_k^M},\\
&\frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}}\mathrm{E_k})}{\partial \mathrm{t}} + \frac{\partial(\alpha_{\mathrm{k}}\rho_{\mathrm{k}}\mathrm{E_k}\mathrm{u_k} + \alpha_{\mathrm{k}}\mathrm{p_k}\mathrm{u_k})}{\partial \mathrm{x}} + p_{ik}\frac{\partial \alpha_k}{\partial t} = \mathrm{S_k^E},
\end{aligned} \tag{1}$$

where subscript k = 0 or 1 indicates the gas or liquid phases and α_{k}, ρ_{k}, u_k, $\mathrm{p_k}$, $\mathrm{E_k}$ represent volume fraction, density, velocity, pressure and total Energy of phase k, respectively. Also, $\dot{m}$ is the mass transfer between the gas and liquid phase due to phase change, $\mathrm{u_{ik}}$ is the interfacial velocity and $\mathrm{p_{ik}}$ is the interfacial pressure. The terms S_k^M and S_k^E are the momentum and total energy source terms which encapsulate the physical effects such as friction, gravity and heat transfer. Term $\mathrm{r_p}$ represents the pressure relaxation parameter which regulates the interaction between two phases. Considering compressible liquid and gas phases, a relevant EOS can be used to complete the governing equations. Now, the system 1 can be reformulated in vectorial form as follows:

$$\frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{A}\frac{\partial \boldsymbol{q}}{\partial x} = \boldsymbol{R} + \boldsymbol{S}, \tag{2}$$

where vector $\boldsymbol{q} = \left[\alpha_g\ \alpha_k\rho_k\ \alpha_k\rho_k u_k\ \alpha_k\rho_k E_k\right]^T$ is the vector of conservative variables and $\boldsymbol{A} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}}$ is the Jacobian matrix of flux function $\boldsymbol{f}$, and the vectors $\boldsymbol{R}$ and $\boldsymbol{S}$ are related to the relaxation and sources terms. This system has always 7 eigenvalues and is unconditionally hyperbolic. Using the fractional splitting method [10], the system 2 could be split into three sub-problems that can be solved independently at each time step. The 1st step (called hyperbolic step) is the homogeneous partial differential equations ($\frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{A}\frac{\partial \boldsymbol{q}}{\partial x} = 0$) which is discretized using 1st order upwind method. It is solved using Roe's method linearization which approximates the solution of the nonlinear hyperbolic system 2 by the exact solution of its linearized counterpart. The description of Roe method is out of the scope of this paper, for more details refer to [11,12]. The 2nd and 3rd steps are simple ordinary differential equations which are integrated explicitly. These steps contain the source terms and the relaxation terms, respectively.

LOOK-UP TABLE METHOD FOR CO_2

SW EOS was originally provided in the density-temperature ($\rho - T$) space, refer to [5]. However, the conservative formulations, such as the one utilized in this study, require the EOS to be supplied in the density-internal energy ($v - e$) space. It is necessary to transfer the SW EOS to the new $v - e$ space before generating the table, see e.g. Fig. 1. This transformation itself is a complex process and need time consuming iterative processes. However, we omit the related discussion here as it is straightforward.

As shown in Fig.2, the areas of both the liquid and gas regions are of similar magnitude within ($T - P$) domain. However, the size of the liquid, gas and two-phase regions vary significantly in this new space ($v - e$). Therefore, a single table with uniform spacing will not provide a consistent accuracy across the full domain of interest. It is important to note that the x-axis in Fig. 2 is

logarithmic, and in practice, the size of the liquid region is several orders of magnitude smaller than the gas region. To tackle this issue, separate tables with different spacing and corresponding conformal mapping have been generated for each region (liquid, gas, two-phase regions) in the present work. These requirements, along with the efficient implementation of the look-up table method, set this approach apart from typical look-up table methods utilized in other traditional flow assurance software.

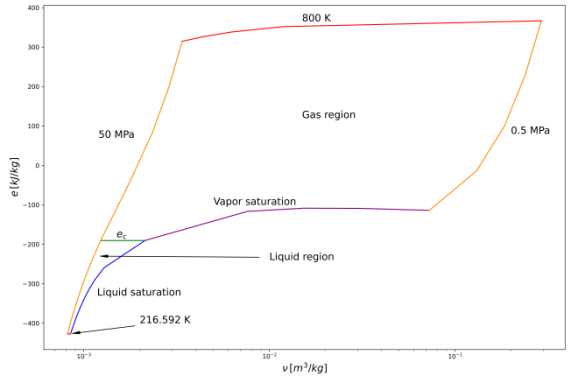

Figure 1. Thermodynamic diagram $v - e$ in the physical domain for CO_2 based on SW EOS.

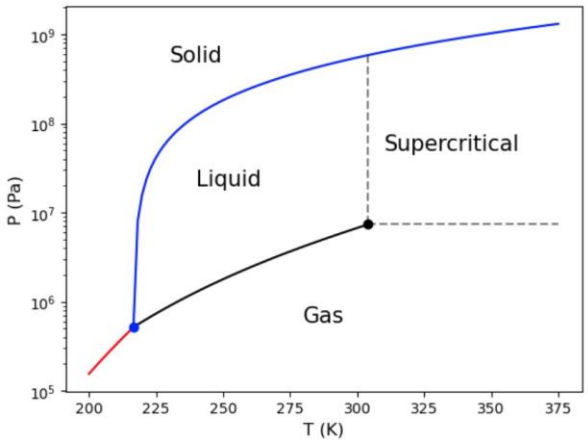

Figure 2. Thermodynamic diagram $T - P$ in the physical domain for CO_2 based on SW EOS.

In the conventional approach, such data is typically provided in tabular format in pressure-temperature ($P - T$) space. During the hyperbolic step, by considering the density and internal energy of each computational cell, it becomes feasible to identify the four nodes in the look-up table that bracket the desired values. Then a bilinear interpolation between these values is employed to obtain the required properties at each cell, e.g. temperature, pressure, sound speed, refer to [6] for more details. Due to the inconsistency in phase partitioning within the two-phase region, it is common practice to avoid using look-up tables for this area. However, in the current study, we utilized the table solely for determining the temperature of the two-phase setting. Subsequently, this temperature was employed to accurately compute the properties of the liquid and vapor phases. Finally, these properties were utilized to calculate the volume fractions of each phase using conservation equations. Through examination, we selected 5000 random points in each region, and calculated the properties of these points, using the look-up table and origin SW EOS. The highest error of our table look-up algorithm is less than 1 percent across all possible configurations (liquid, vapor and liquid-vapor regions). Additionally, the computational overhead of the look-up table is negligible (compared to SG EOS) in single-phase flows and is significantly faster than two-phase flows modeled with simplified equations of state such as NASG, particularly when phase change is present.

GPU Acceleration

GPUs with their many-core architectures and higher memory bandwidth, has become one of the most important parts of high performance computing. Our explicit numerical method, with its small (local) discrete stencil and its time marching nature, can employ the power of the GPUs very efficiently which can accelerate the simulations tremendously. In the present work, our numerical method is fully implemented on the GPU using CUDA. In this implementation, all data are stored on the GPU's memory to minimize the data transfer between CPU and GPU, and the subsequent calculations are performed directly on the GPU as well. In the data transfer between GPU and CPU (read/write operations), the copy process is overlapped with kernel launches. It has been tried to use the shared memory as much as possible and also maximize coalesced access to the global memory where possible. The 1D nature of the simulations enables us to efficiently accomplish these tasks. Our GPU implementation of the present numerical solver demonstrates a two-order-of-magnitude speedup compared to the typical multi-core CPU implementation (NVIDIA V100 GPU vs. 8 cores Intel I9 CPU).

Results and Discussion

To evaluate the accuracy and performance of the developed look-up table, we performed different simulations of benchmark test cases, reported in literature, such as a shock tube, depressurization in a pipe. As an example, a classic shock tube problem is considered here. It is a 100-m tube which is filled with CO_2 at two different vapor states with a membrane at the middle of the tube. The left state is at $p_L = 3\,MPa$, $T_L = 300\,K$ and $\rho_L = 63.376\,kg.m^{-3}$ while the right state $p_R = 1\,MPa$, $T_R = 300\,K$ and $\rho_L = 18.579\,kg.m^{-3}$ at rest. By removing the membrane at $t = 0$, The left side with higher pressure interacts with the right side. The expansion wave propagates toward the left side while the shock wave and contact discontinuity propagate toward the right side. Due to this initial condition, the liquid and gas phases are in equilibrium in both left and right sections of

the shock tube. However, the generated waves could alter the state of the matter, leading to phase change. Our numerical method effectively captures these effects. Figure 3 shows the pressure distribution along the tube at $t = 0.08\,s$. As seen, our results match very well with those reported in Giljarhus et al. [11].

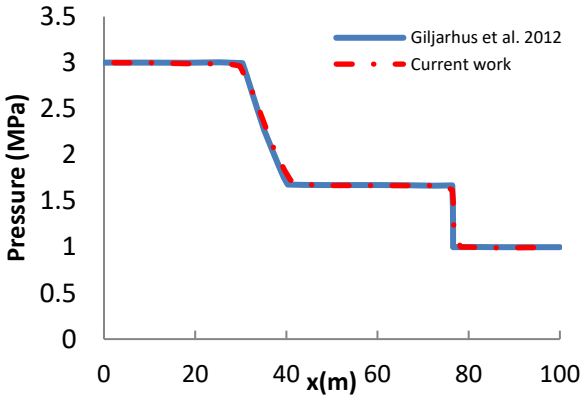

Figure 3. Pressure profile along the shock tube at $t = 0.08\,s$, in comparison with the numerical results reported in Giljarhus et al. [13]

Conclusions

The unconditionally well-posed 7-equation slug capturing solver has been developed in this paper to numerically simulate two-phase compressible CO_2 flows. We have successfully developed a look-up table method tailored for our solver, leveraging the original Span-Wagner EOS. This approach compromises between high accuracy and reduced computational costs, addressing a critical need in simulating two-phase compressible CO_2 flow for applications such as carbon capture, storage and utilization (CCSU). By using a GPU-based solver, we have achieved substantial speedup, further enhancing the efficiency of our methodology. Our results demonstrate that the predictions obtained using the look-up table method are remarkably close to those derived from the original Span-Wagner EOS, while requiring significantly fewer computational resources. Our future work will be devoted to extending this work into a many-phase solver, where CO_2 (liquid, gas, or two-phase) and impurities are present in the flow. Also, we will implement other state-of-the-art EOSs such as GERG, CG and CPA, in the same framework.

Acknowledgments

The authors acknowledge the technical and financial support of Petrogal Brasil S.A. (Joint Venture Galp | Sinopec) and the promotion of Research, Development and Innovation (R, D & I) in Brazil by the National Agency of Petroleum, Natural Gas and Biofuels (ANP) for the execution of this project.

Responsibility Notice

The authors are the only responsible for the paper content.

References

[1] Xu, B.; Jie, H.; Wen, J. X.; Cooper, R.; Barnett, J., IChemESymp. Ser. No. 156, p. 22, 2011.
[2] Morin, A.; Aursand, P. K.; Flåtten, T.; Munkejord, S. T., Society for industrial and applied mathematics, pp. 108-119, 2010.
[3] Raman, S. K.; Kim, H. D., Int. J. Heat mass transfer, 116, 127-135, 2018.
[4] Aidoun, Z.; Ouzzane, M., Int. J. Refrig., 27, 974-984, 2004.
[5] Span, R.; Wagner, W., Journal of physical and chemical reference data, 25 (6), 1509-1596, 1996.
[6] Fang, Y.; De Lorenzo, M.; Lafon, P.; Poncet, S.; Bartosiewicz, Y., Industrial and engineering chemistry research, 57 (22), 7676-7691, 2018.
[7] Nikfarjam F.; Anbarlooei, H; Cruz, D. O. A., Multiphase Flow Dynamics: A Perspective from the Brazilian Academy and Industry, pp. 199-210, 2022.
[8] Nikfarjam, F.; Anbarlooei; H., Cruz D. O. A.; Dresjan Junior, C., SPE Brazil Flow Assurance Technology Congress, Rio de Janeiro, 2022.
[9] Ferrari, M.; Bonzanini, A.; Poesio, P., International journal for numerical methods in fluids 85, 327-362, 2017.
[10] Saurel, R.; Abgrall, R., J. of computational physics, 150, 425-467, 1999.
[11] Roe P. L., Journal of computational physics, 43(2), 357-372, 1981.
[12] Toumi I., Journal of computational physics, 102 (2), 360-373, 1992.
[13] Giljarhus, K. E. T.; Munkejord, S. T.; Skaugen, G., Ind. Eng. Chem. Res. 51, 1006-1014, 2012.