21st Annual CFD Symposium, August 8-9, 2019, Bangalore

Implementation and application of AUSM+ -up and AUSM+ -up2 schemes in open-source CFD code SU^2

Amit Sachdeva
Aerodynamic Configuration Design Division
Vikram Sarabhai Space Centre, Thiruvanthapuram, India.
(amit_sachdeva@vssc.gov.in)

Abstract

In the present work, efforts have been made to enhance the capability of open-source CFD code SU^2 for low as well as high speed flows. AUSM+ -up and AUSM+ -up2 schemes have been implemented in the code. These schemes feature many improvements over their predecessors in terms of accuracy and robustness. Simulations have been carried out to test and validate the implementations. Results from these schemes are compared with various existing numerical schemes in the code. Analysis shows improvement in accuracy for low speed flows as well as good results for high Mach number flow conditions. This work also reflects contribution towards improving the capabilities of open-source CFD code for benefit of the community.

Keywords: SU^2 , AUSM, Roe, HLLC, SLAU

Nomenclature

M	Mach number	AUSM	Advection Upstream Splitting Method
α	Angle of attack	SLAU	Simple Low Dissipation AUSM
C_p	Pressure coefficient	SA	Spalart Allamaras Turbulence model
SU^2	Stanford University Unstructured	SST k-ω	Shear Stress Transport k-ω Turbulence model

Introduction

Open-source codes are finding increasing role in engineering and scientific community in various areas. There are many advantages associated with open-source codes, primary of which are free license/unlimited usage and full source code access. They also provide a platform for larger community contribution from across the globe. There are issues associated with them as well, for example quality control, performance related aspects etc. in comparison to their commercial counterparts. There are many open-source CFD codes available today. Some of them have come to maturity level for industrial usage and some of them are rapidly heading in that direction. SU^2 (Stanford University Unstructured) [1] code is one of the rapidly growing open-source CFD code. Many advancements have taken place in the code recently and various features are being added regularly by the growing community across the globe. In the present work, couple of convective numerical schemes are added for further improving the capabilities of SU^2 CFD code.

SU^2 CFD code

SU^2 primarily uses Finite Volume Method (FVM) based framework (median dual cell vertex based scheme) for discretization of Euler and Navier-Stokes equations. It is primarily designed for compressible flow applications. It can handle structured, unstructured grids written in SU^2 native mesh format as well as CGNS format. It has options of explicit, implicit time integration for steady state computations and dual time stepping for unsteady flow computations. For RANS simulations, SA and SST k-ω turbulence models are available. These are some of the features of SU^2 code among many others. One of the key feature of SU^2, which sets it apart from other codes is adjoint solver (continuous and discrete) for shape optimization and other purpose. Various new capabilities are being regularly added to the code, some of the recent additions are Fluid Structure Interaction (FSI), Detached eddy simulation (DES) etc.

Implementation of AUSM+ -up / AUSM+ -up2 numerical schemes

In the present work, AUSM+ -up and AUSM+ -up2 [2, 3] numerical schemes have been added in the code. AUSM scheme was developed by Liou M-S. [4]. It has many attractive features and is very popular in CFD community. This scheme has been later improved by Liou and many other researchers, which includes applications to chemically reacting, multiphase flows etc. AUSM+ -up scheme features enhancements for low speed computation over its predecessor AUSM+ [5]. Here pressure diffusion term is added in interface Mach number definition and velocity diffusion term is added to pressure flux definition. These terms reduce the excessive numerical dissipation by properly scaling it in low Mach number limit. Also this scheme is less carbuncle prone for high speed flows in comparison to its predecessors.

AUSM+ -up2 scheme is developed by Kitamura K. The difference in this scheme is that pressure flux definition of AUSM+ -up scheme is replaced by SLAU2 pressure flux definition. Here, numerical dissipation not only scales for low Mach number limit but scales proportionally with Mach number for M>1. Hence more dissipation is added with increasing Mach number, which is likely to improve the robustness of the scheme against various shock related anomalies.

SU^2 code is written in C++ language and leverages the strengths of Object Oriented Programming framework, which allows re-usability of existing methods and flexibility of adding new capabilities with relative ease. Along with C++, it also uses python for optimization, multi-physics simulations etc. More details about the code structure can be found in the main reference paper [6]. SU^2 code structure allows for fast and clean implementation of the new methods.

Two new classes were added in the code, namely **CupwAUSMPLUSUP_Flow** and **CUpwAUSMPLUSUP2_Flow** for these schemes along with the functions for computing the residuals for the same. These contributions have been released in SU2-6.2.0 .

Results and discussion

Few test cases have been carried out with the new schemes and results are compared with various existing convective numerical schemes to assess the implementations.

I) - NACA 0012 airfoil

(a) Inviscid flow past NACA 0012 airfoil (M=0.01, 0.1 at $\alpha=0^o$)

Inviscid flow simulations have been carried out over NACA 0012 airfoil at Mach number 0.01, 0.1 for 0^o angle of attack. Second order accurate simulations have been carried with Venkatakrishnan limiter, and implicit method (FGMRES solver with LU-SGS preconditioner) is used for the time integration. Unstructured mesh available in SU^2 test case repository has been used for this case. Tests are carried out with various numerical schemes (Roe, HLLC, SLAU, SLAU2, AUSM+ -up, AUSM+ -up2) and results are compared.

Figure 1 presents the pressure contours in the stagnation region of the airfoil. It can be observed that Roe and HLLC scheme produce unphysical values and spurious contours in the stagnation region, while AUSM+ -up and AUSM+ -up2 produce smooth contours with more accurate stagnation C_p values (closer to 1). Open-source tool ParaView [7] has been used for post-processing work. Figure 2 shows pressure contours comparison for Roe and AUSM+ -up scheme. It can be clearly observed that, AUSM+ -up produces much smoother and accurate flow-field. Figure 3 shows comparison of pressure distribution over the airfoil with various schemes. Roe and HLLC schemes produce high C_p value locally in the stagnation region while AUSM+-up/up2 and SLAU/SLAU2 produce correct variation. Figure 4 presents comparison of pressure distribution for Mach number 0.1. Here Roe and HLLC schemes produce slightly higher stagnations C_p value (5%), while for SLAU (2) and AUSM+ -up(2) stagnation C_p is very close to 1 (within 1%).

One observation with these simulations was that allowable CFL for AUSM+ -up/up2 schemes was around 0.1 (for M=0.01 case), which is very small. This resulted in very large number of iterations for these two schemes before obtaining the converged results. One of the possible reason could be inconsistent Jacobian (Roe Jacobian) used in implicit time integration for these specific schemes. This suggests use of consistent Jacobian to improve the allowable CFL. Also it should be noted that there is no other specific technique like time derivative preconditioning is used, addition of which also may likely improve the convergence rate.

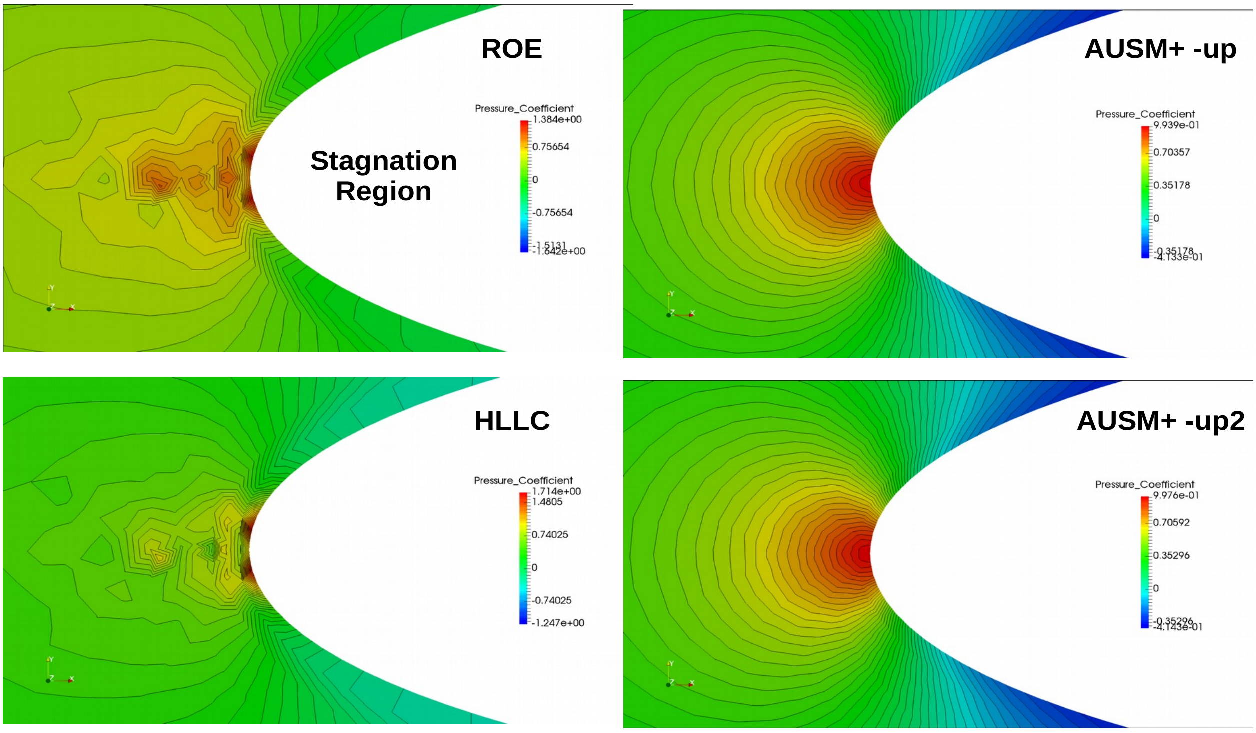

Figure-1 C_p palette/contours in stagnation region of NACA 0012 airfoil (M=0.01)

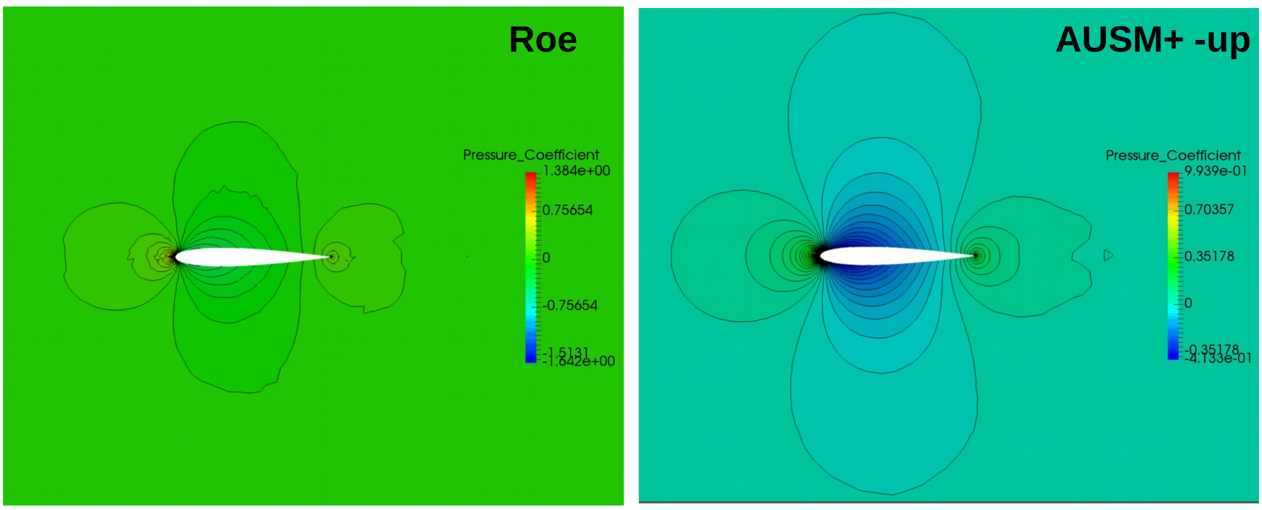

Figure-2 Cp contours over NACA 0012 airfoil (M=0.01)

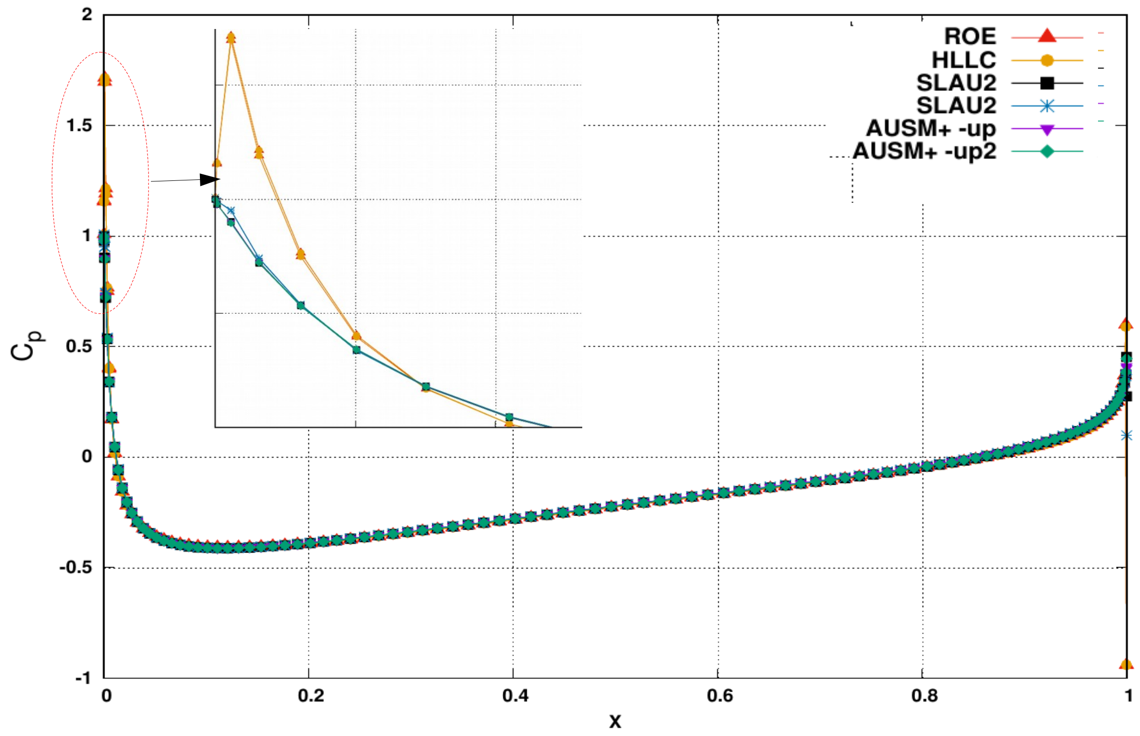

Figure-3 Comparison of C_P variation for various schemes over NACA 0012 airfoil (M=0.01)

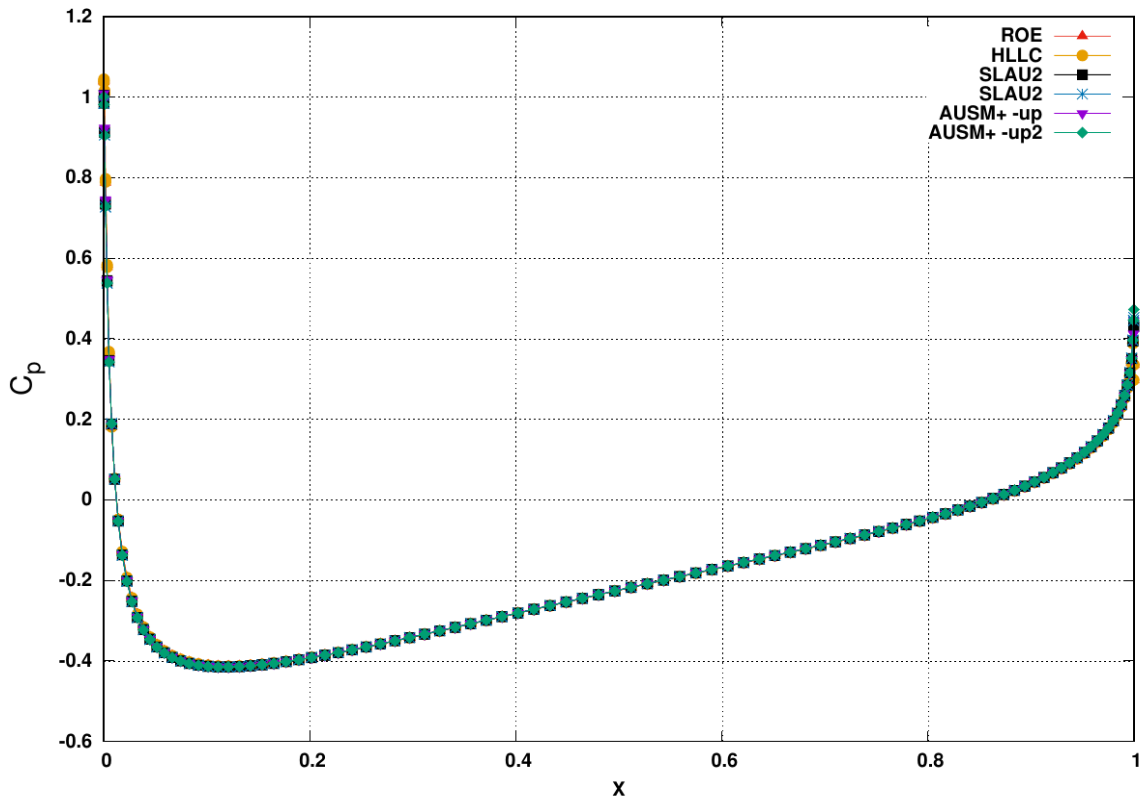

Figure-4 Comparison of C_P variation for various schemes over NACA 0012 airfoil (M=0.1)

(b) Inviscid flow past NACA 0012 airfoil (M=0.8, α=1.25º)

Another inviscid flow simulation test case is carried out at transonic Mach number 0.8, angle of attack 1.25º. Here a strong normal shock forms over the leeward side and a weaker shock forms in the windward side. Figure 5 shows pressure contours with AUSM+ -up and SLAU2 schemes. It can be seen that shock is captured more crisply with AUSM+ -up in comparison to SLAU2 scheme, pointing out comparatively lower numerical dissipation for AUSM+ -up scheme. This can be seen in Figure 6 as well, where comparison of pressure distribution with AUSM+ -up/up2 and SLAU2 schemes is shown.

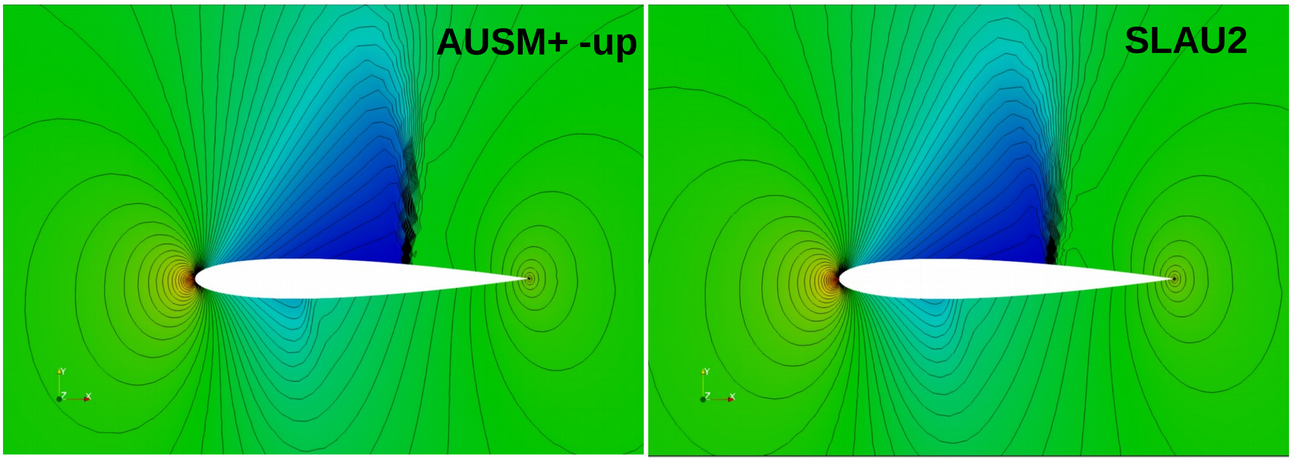

Figure-5 C_p contours over NACA 0012 airfoil (M=0.8, α=1.25º)

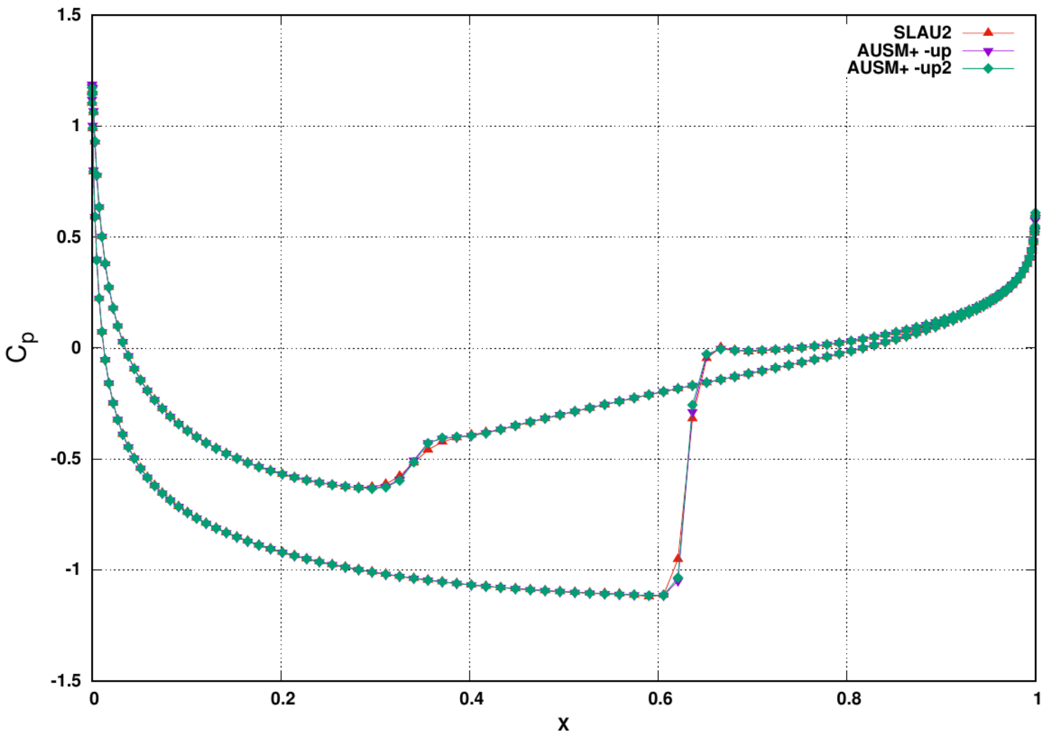

Figure-6 Comparison of C_P variation over NACA 0012 airfoil (M=0.8, α=1.25º)

II) - Hypersonic Inviscid flow past blunt body (M=6.0)

To assess the performance of the schemes for high speed flows, first order accurate hypersonic inviscid flow simulations have been carried out over blunt body at Mach number 6.0. A structured grid with dimensions 35 x 91 has been used for the simulations. Implicit time marching with constant CFL value of 3.0 has been used. Simulations have been carried out with various schemes (Roe, HLLC, SLAU2, AUSM+ -up and AUSM+ -up2) and results are compared.

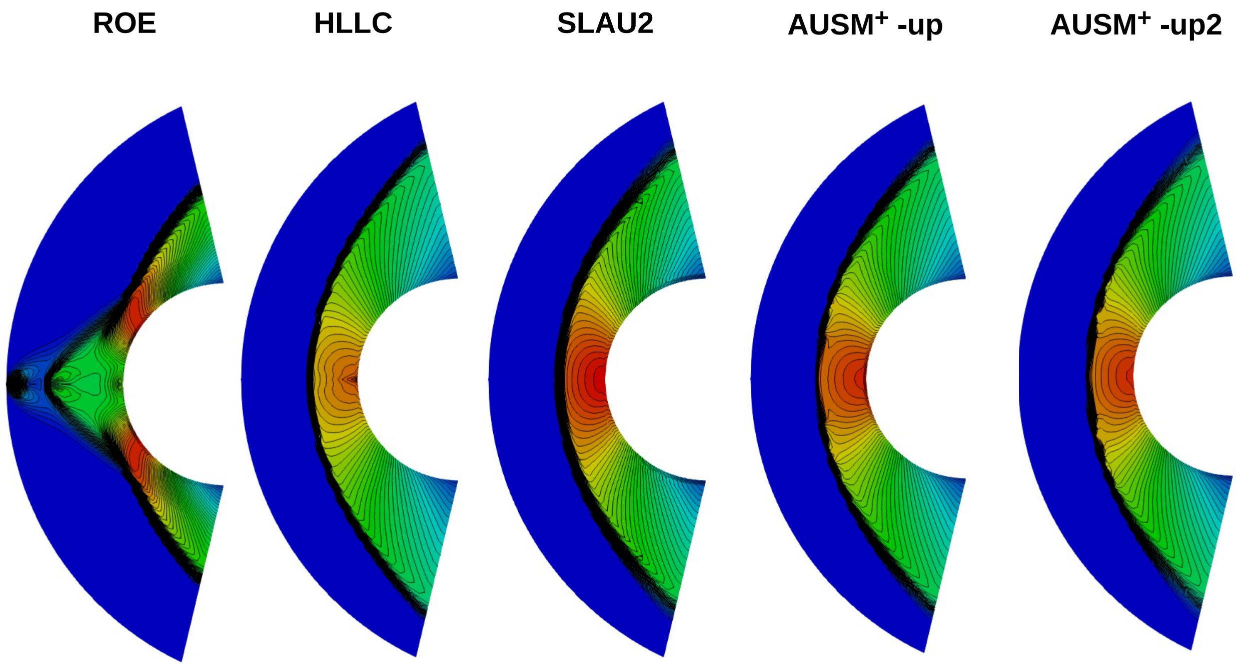

Figure-7 C_p palette/contours over blunt body in hypersonic flow (M=6.0)

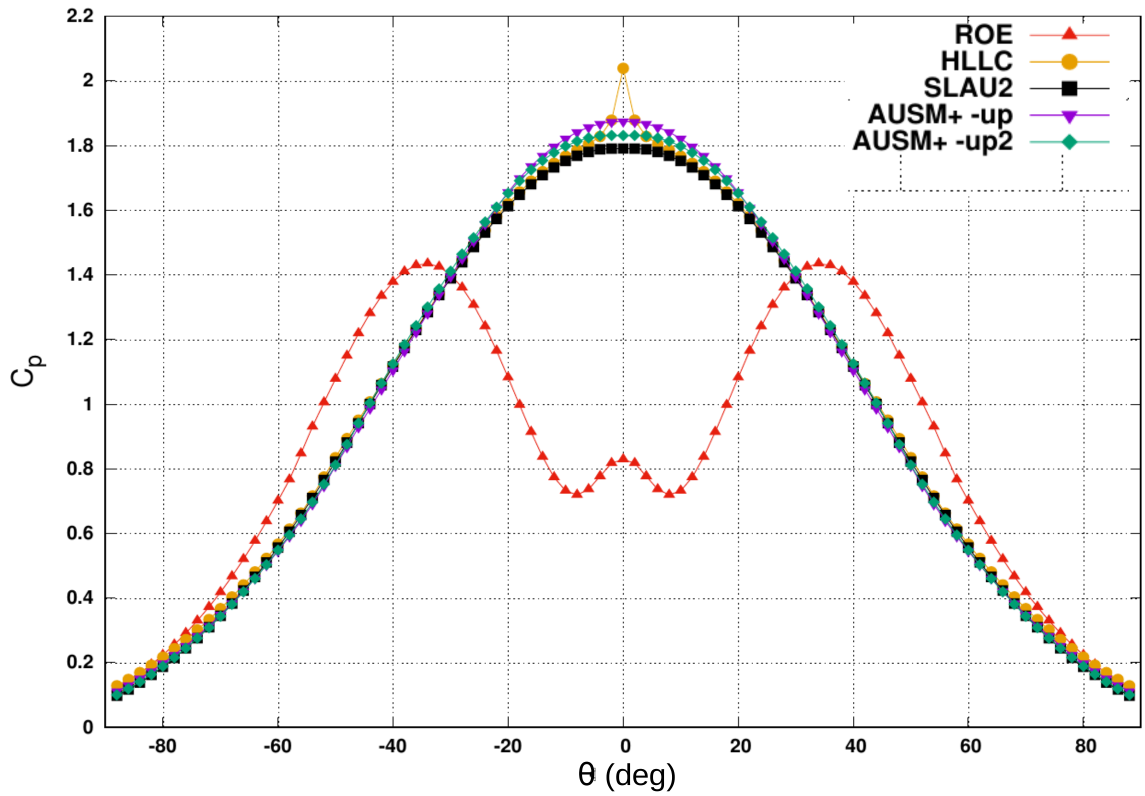

Figure-8 Comparison of C_p distribution for various schemes over NACA0012 airfoil

Figure 7 shows comparison of pressure contours with various schemes and Figure 8 gives comparison of pressure distribution for the same.

Following observations can be made from the above results -

- Roe scheme clearly exhibits carbuncle phenomenon in the stagnation region.
- HLLC scheme produced small kink in the stagnation region for this particular case.
- SLAU2, AUSM+ -up/up2 show smooth pressure contours and variation in and close to the stagnation region, which is evident from the pressure variation plot as well.
- Shock captured by AUSM+ -up / up2 are slightly more crisp than SLAU2.

For this grid and conditions, SLAU2 and AUSM+ -up/up2 produced smooth pressure variation (no carbuncle), which reflects their relatively robust nature in comparison to other schemes.

Conclusion and Future work

AUSM+-up and AUSM+-up2 numerical schemes have been implemented in open source CFD code SU^2. Numerical simulations have been carried out to validate and assess the implementation of the schemes. Results are compared with various existing numerical schemes in the code to demonstrate the improvements in accuracy as well as robustness.

In the present release of the code (SU2-6.2.0), Roe Jacobian is used for the implicit part for AUSM+ -up/up2 schemes, which results in inconsistent discretization and possibly leads to lower allowable CFL than probably could have been achieved with the consistent Jacobian. Future work is planned in this direction to further improve the performance of these schemes with consistent Jacobian for implicit time integration.

Acknowledgement

I would like to acknowledge Dr. Pankaj Priyadarshi and Dr. Dipankar Das, VSSC for their suggestions and feedback. I also acknowledge Dr M. M. Patil and Dr. V Ashok, VSSC for their help and support. I would also like to thank Dr. Thomos D. Economon, Dr. Eduardo Molina and Dr. Padro from Stanford university for their guidance and suggestions for the above work.

References

1- Palacios F, Colonno M, Aranake A, Campos A, Copeland S, Economon T, Lonkar A, Lukaczyk T, Taylor T and Alonso J, "Stanford University Unstructured (SU^2): An open-source integrated computational environment for multi-physics simulation and design", 51 st AIAA Aerospace Sciences Meeting, 2013, AIAA 2013-0287.

2- Liou M-S. "A sequel to AUSM, part II: $AUSM^+$ -up for all speed. Journal of Computational Physics 2006"; 214:137–170

3- Kitamura, K. and Shima, E.: "Towards shock-stable and accurate hypersonic heating computations: A new pressure flux for AUSM-family schemes", Journal of Computational Physics, Vol.245 (2013), pp.62-83

4- M.-S. Liou, C.J. Steffen Jr., "A new flux splitting scheme", Journal of Computational Physics 107 (1993) 23–39. Also NASA TM104404, May 1991.

5- M.-S. Liou, "A sequel to AUSM: AUSM +" , Journal of Computational Physics 129 (1996) 364–382. Also NASA TM 106524, March 1994.

6- T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, and J. J. Alonso. "SU2: An open-source suite for multiphysics simulation and design." AIAA Journal, 54(3):828–846, 2016.

7- ParaView Manual, Version – 5.0