FEMZIP COMPRESSION AND FASTER METAPOST VISUALIZATION OF CFD RESULTS

[1]Pinaki Banerjee*, [1]Stefan P. Müller, [2]Giorgos Kalaitzidis, [2]Dimitrios Siskos
[1]SIDACT GmbH, Germany, [2]BETA CAE Systems SA, Greece

KEYWORDS
Compression, Simulation Results, Reduced Data Storage, Faster Post-Processing

ABSTRACT
Computer simulations, as commonly used for product development in various industries, generate a vast and growing amount of data. The growth in data is a result of larger, more detailed models and an increase in the volume of simulations performed to improve engineering design. Simulation data has to be analysed, exchanged among engineers and archived for future reference and reanalysis. Network connections and storage space can become bottlenecks in workflows used by engineers. The challenge is to handle large amounts of data in time and storage saving manner in order to eliminate these bottlenecks.

The difficulties can be approached by using FEMZIP compression methods. These tools are specifically designed for the compression of Finite Element Method and Computational Fluid Dynamics (CFD) based simulation results and thus achieves high compression factors. The significant reduction in data volumes leads to a reduction in the requirements for storage and faster data exchange over networks.

To improve the usability of compressed datasets, FEMZIP compressed output format is always designed based on the original format structure. Thus resulting in easy integration into post-processors and on-the-fly access of data subsets without the requirements of original data reconstruction

As a part of our latest developments we have implemented our first generation of CFD data compression for OpenFOAM format. The post-processing of compressed data has been already integrated into MetaPost and the results will be illustrated in the following.

TECHNICAL PAPER

1. INTRODUCTION

The software tool FEMZIP is well known for the compression of data sets arising in the context of crash test simulations and noise vibration harshness (NVH)-simulations. As a new application field of our techniques, we now support the compression of OpenFOAM simulation results. For transient data, we achieve a compression factor of more than 10 with a lossy compression. Hereby, the engineer has the possibility to set a desired precision that will strictly be adhered to.
A reduced data size has several advantages. The one we will set our focus on is the reduced read in time of a FEMZIP compressed data set compared to the read in time of data sets in uncompressed OpenFOAM format.

2. ON-THE-FLY COMPRESSION

Since data sets in CFD are multiple times bigger than in crash tests or NVH simulations, we adapted our strategy to be able to compress simulations on-the-fly. This will reduce the traffic between clusters and servers drastically.

For this purpose, we created a FEMZIP Daemon that is running parallel to the OpenFOAM processes and checks from time to time if a time step is written out completely. If one time step is finished, a FEMZIP process will be started and compress the time step by exploiting dependencies in time and space. When the compression ended successfully, the compressed data set will be appended to one compressed file. Therefore, the compressed file will not be corrupted if simulations fail.
Usually, OpenFOAM cases are simulated in parallel. We exploit the decomposition into processor files by an application of several FEMZIP processes that will perform the compression in parallel.

3. RESULTS

We investigate the compression and read in times of a data set of a big OEM with 10 time steps, four variables, about 60 Mio nodes and 55 Mio cells.

Compression rates

The compression rates for the geometry that is containing the topology of the mesh and the coordinates of the nodes can be found in Table 1.

Size geometry	6,293,824,952
Size compressed geometry	952,620,292
Compression ratio	**6.61**

Table 1 – Compression factors and sizes in Bytes of the geometry of an OEM model.

The compression rates for the variables velocity (U), pressure (p), nuSgs and nuTilda can be found in Table 2.

Variable name	U	nuSgs	nuTilda	p	Sum
Size originally	14,448,836,952	3,006,049,312	3,956,520,312	4,816,170,392	2.6228E+10
Size compressed	1,170,940,165	13,130,566	2,331,096	506,524,554	1,692,926,381
Compression ratio	**12.34**	**228.94**	**1697.28**	**9.51**	**15.49**

Table 2 – Compression factors and sizes in Bytes of the variables of an OEM model.

In Table 3, you can find the accumulated sizes and the overall compression rate for the transient model.

Size originally	32,521,401,920
Size compressed	2,645,546,673
Compression ratio	**12.29**

Table 3 – Overall compression rates and sizes in Bytes of an OEM model.

The results show that the size of the data can be reduced by a factor of more than 12 in respect to precisions that are approved by the engineers.

Visualization in MetaPost 15.3.0

In Table 4, we listed the timings for the visualization of the geometry and scalar variables in four test cases. In the first case, the data is stored locally on an HDD. In the second case, we have the data we want to read in locally on an SSD. In the third and fourth case we read the data over a network connection with 10 Gigabit per second (Gbps) and 1 Gbps, respectively. For Table 4 the processor addressing information is available in the data set. This cannot be ensured for all data sets. Therefore, we investigated the read in times for the case where no procAdressing information is available in Table 5.

	FEMZIP	BINARY	BINARY GZIP
Local HDD : Geometry	**70**	148	110
Local HDD : Scalar	**85**	116	113
Local SSD : Geometry	**49**	71	94
Local SSD : Scalar	**55**	**55**	87
Network 10Gbps : Geometry	**50**	72	109
Network 10Gbps : Scalar	**52**	**52**	86
Network 1Gbps : Geometry	**56**	115	110
Network 1Gbps : Scalar	**60**	72	101

Table 4 – Read-in times in seconds for FEMZIP compressed OpenFOAM data and the original OpenFOAM data sets with procAdressing information.

	FEMZIP	BINARY	BINARY GZIP
Local HDD : Geometry	**75**	160	119
Local HDD : Scalar	**79**	105	108
Local SSD : Geometry	**54**	75	102
Local SSD : Scalar	**49**	**49**	81
Network 10Gbps : Geometry	**55**	73	104
Network 10Gbps : Scalar	**48**	**48**	83
Network 1Gbps : Geometry	**62**	121	113
Network 1Gbps : Scalar	**56**	68	97

Table 5 – Read-in times in seconds for FEMZIP compressed OpenFOAM data and the original OpenFOAM data sets without procAdressing information.

When we investigate a merged data set we get the timings of Table 6.

	FEMZIP	MERGED BINARY	MERGED BINARY GZIP
Local HDD : Geometry	**62**	96	216
Local HDD : Scalar	**56**	66	99

Table 6 – Read-in times in seconds for FEMZIP compressed OpenFOAM data and the merged OpenFOAM data sets without procAdressing information.

4. CONCLUSIONS

On-the-fly compression can solve several problems that occur in handling big data sets of many Gigabytes up to several Terabytes. The traffic between servers and clusters can be reduced drastically and the size of the scratch on the cluster can be kept small. Moreover, the data compression can improve the speed of visualization by an optimized integration of our decompression library in MetaPost 15.3.0.

REFERENCES

(1) FEMZIP, SIDACT GmbH, http://www.sidact.com/femzip.html, June 2015
(2) OpenFOAM, The OpenFOAM foundation, www.openfoam.com, June 2015
(3) μETA PostProcessor version 15.3.0, BETA CAE Systems S.A., June 2015