Tool cooling simulation for hot forming

Takeshi KUROIWA

JSOL Corporation

1 Introduction

To fulfill recent regulations for automobile fuel economy there is a growing demand on saving weight of automobiles. Since making a lighter car with conventional material loses occupant safety, at least stiffer materials with the same weight are needed. For example, use of CFRP (Carbon Fiber Reinforced Plastics), Aluminum, Magnesium or Titanium is attracting our attention in these days. But technique to handle these materials is still under developmental stage. High-tensile steels made by hot forming is one of the most promising candidate since it can realize better balance between cost and weight saving.

In hot forming technique, heated blank material is pressed by tools and then quenched by various methods to cause martensitic transition of the blank to obtain high tensile steel. It is not only stiff but also has good shape freezing property, causing smaller springback of the stamped materials. As an another advantage of hot forming, steels as raw materials can be easily obtained all over the world, compared with other materials listed above. On the other hand its disadvantages is relatively large investment in plant and equipment such as chiller or cooling tower and costs for prototyping production of tools with pipes to run cooling water.

In order to cause the martensitic transition of the blank materials, one needs to quench it sufficiently fast. We, JSOL, think that a CAE tool to calculate and predict the stiffness of high-tensile materials contributes to ensure their strength in mass production stage. Important points in accurate prediction are following: (i) to calculate phase transition of the materials correctly (ii) to predict cooling performance of the tools to ensure (i). By making these uncertainties clear it is expected that CAE is capable of reducing trials-and-errors on prototyping, causing reduction of tool designing costs.

LSTC, Dynamore and JSOL have been working on formulating a manufacturing CAE solution to the hot forming techniques. For example, development of phase transition material models (*MAT_244, *MAT_254) will overcome the uncertainty (i) described above. We also have been investigating simulation technique for thermal-structural-fluid coupling calculation to demonstrate the behavior of cooling water flowing in pipes of the tools, corresponding to item (ii) above. In this paper we report the results of recent solution developments on the latter point.

2 Motivation

Our goal is twofold. First is to predict cooling ability of tools when their CAD data is given. Second is to provide a designer's CAE tool to predict the distribution of martensitic phase ratio of blank material for multi-shot process, as a result of accomplishment of the first goal. To achieve these, we have to calculate correctly deformation of work material, temperature of work, die and cooling water, together with velocity and pressure of the water. Such problem is classified to the multiphysics problem, which should be tackled by next generation CAE. As a first step toward our goal we here only consider tools and cooling water and so far ignore the existence of the workpiece.

In order for the simulation of tool cooling to reduce the number of trial-and-error in prototyping of the tools, its calculation time should be much shorter than the time needed to create tools. To this end we have to minimize the total hours from creation of CAD data to the estimation of cooling performance of the tools. It can be divided into the time to prepare the simulation data and calculation hours. Here we put our focus on the latter.

In general fluid velocity and pressure, which characterize the motion of the fluid, equilibrate to their steady state much earlier than temperature of itself or of solid. In other words there is a separation of

the timescale between motion of the fluid and heat transfer. Or it can be also said that a multiscale nature of the system is caused by multi-physical nature of the problem. One can notice that there is a possibility to speed up the computation time of the conjugate heat transfer analysis of such kind, by focusing on this nature. Actually the velocity and pressure need not to be updated after their steady states have been reached and only calculation on the temperature of fluid and solid is needed. Since in many situation hot spot of the conjugate heat transfer problem is update of fluid motion, it is possible to speed up the calculation by stop updating pressure and velocity after their steady state, without losing validity of the solution. In the present report speed gain by above item together with other recent development of LS-DYNA is described. We left the validation of the result to the next step.

3 Model

We prepared a benchmark model of simple channel flow with conjugate heat transfer across surrounding structural mesh that has similar load profile distribution as a typical A-pillar tool cooling analysis. We show its number of nodes and elements for both structural and fluid mesh in Table 1.

Table 1 Summery of number of node and elements for the benchmark model used in the papar. Structual mesh is hexagonal, while fluid mesh is consist of tetrahedral elements.

	# nodes	*# elems*
Structure	12,240	10,000
Fluid	262,152	1,508,822

Physical quantities used here are summarized in the table below. Note that because it is just a benchmark model, values and units do not have any meanings. We here employed MAT_RIGID for material of the solid.

physical quantity[unit]	*value*
density of fluid[kg/m^3]	1.0
viscosity of fluid[Pa s]	1.0e-4
heat capacity of fluid[J/(kg K)]	1.0e+5
heat transfer coefficient of fluid[W/(m K)]	2.0
density of structure[kg/m^3]	1000.0
heat capacity of structure[J/(kg K)]	1000.0
heat transfer coefficient of fluid[W/(m K)]	200.0

Boundary conditions and initial conditions for structure and fluid are shown below.

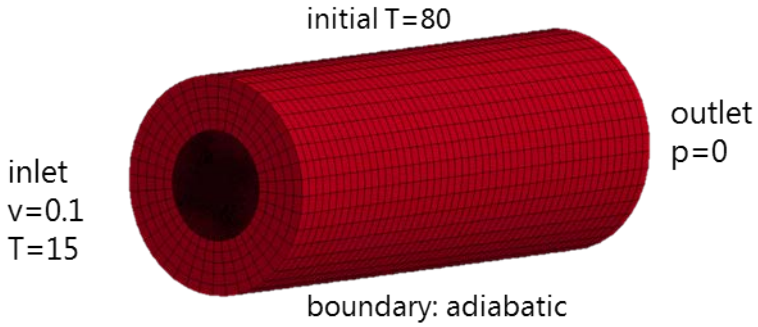

Figure 1 Boundary and initial conditions for structure and fluid mesh.

We carried out 100 step calculation with timestep for thermal, structural and fluid being unity and monitored calculation time reported in the d3hsp file. 5 samples are taken for each calculation and typical data for load profile distribution and calculation time is reported. Calculations are carried out with Intel(R) Xeon(R) CPU E5-2670 @ 2.60GHz, 16MPP inner-node parallel execution. Memory installed per node is 64GB. Module used is development version of mppdyna linux double precision. Revision number is reported at each subsection.

4 Result

In this section we present various speed up item on conjugate heat transfer analysis available on recent development version together with their actual effects. With the model described in section 3 we obtained 13 times speed up in total, compared with a base model. Figure 2 shows cumulative graph of CPU time reported in "T i m i n g i n f o r m a t i o n" section in d3hsp file generated by LS-DYNA. Left bar is for our base model. The hot spot of the analysis is calculations classified into "Thermal", namely, time spent by temperature calculation including matrix assemble for temperature of structural part and linear equation solver for monolithic temperature calculation for whole part (fluid and structure). Next biggest is Incompressible CFD(ICFD) solver. It contains time spent by update of pressure and velocity of the fluid and matrix assemble of temperature of fluid side. Most of the computation time is dominated by thermal calculation in the base model. In the following subsection we one by one present an effect of each speed up items.

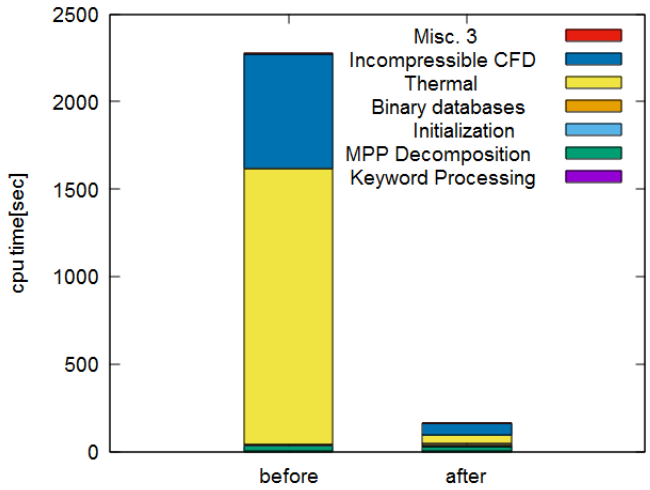

Figure 2 comparison between calculation time for original model and current model.

4.1 Speedup by GMRES solver

Users can select GMRES solver as an iterative solver for monolithic coupling of temperature for a conjugate heat transfer problem with LS-DYNA R9.1.0 or more recent. It is effective especially for a problem with 1M elements or more. The solver can be turned on by setting SOLVER=17 in *CONTROL_THERMAL_SOLVER keyword. As shown in Figure 3, about 2.5 times speed up is achieved compared with the base model, which employs SOLVER=12(diagonal scaling conjugate gradient solver) as default. As a result of this speed up item now hot spot of this analysis switched to ICFD.

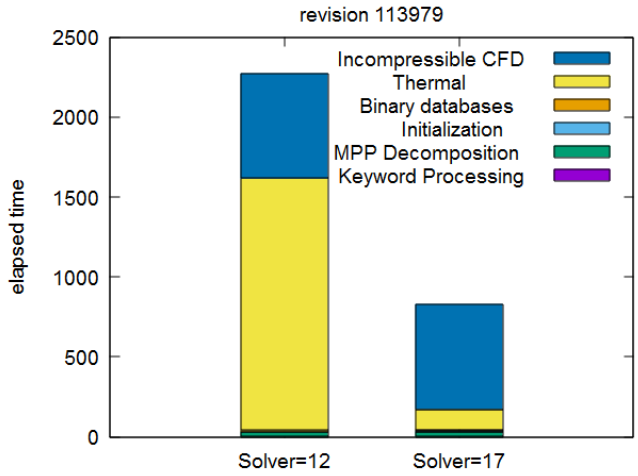

Figure 3 Comparison of calculation time between default thermal solver(solver=12 in *CONTROL_THERMAL_SOLVER) and new GMRES solver(solver=17)

4.2 Speedup by reduction of redundant output calculation

For massively parallelized computation sometimes unparalleled part becomes a hot spot of the computation. Recently such kind of part in ICFD calculation had been resolved and as a result good scalability is achieved. We recommend users who are not satisfied with scalability of ICFD to use development version of mppdyna revision 114684 or more recent. Note that the speed up in parallel execution is available only with MPP module and not with SMP. Figure 4 shows comparison of CPU time for each revision. In between 114428 and 114513 a slight speed up due to omission of calculation of output quantities for d3plot when it is not needed. In between 114513 and 114684 ICFD load is reduced to half of original one. It can be understood by looking into breakdown chart for ICFD shown in the right panel of Figure 4. A detailed profile is available from revision around 114428. Clearly Misc. in ICFD is dramatically reduced in between 114513 and 114684. Note that it is corresponding to the unparalleled and redundant calculation related with certain kind of output. For the users of older version modules this redundancy can be avoided by indicating DTOUT in *ICFD_CONTROL_OUTPUT explicitly different from zero and as large as possible.

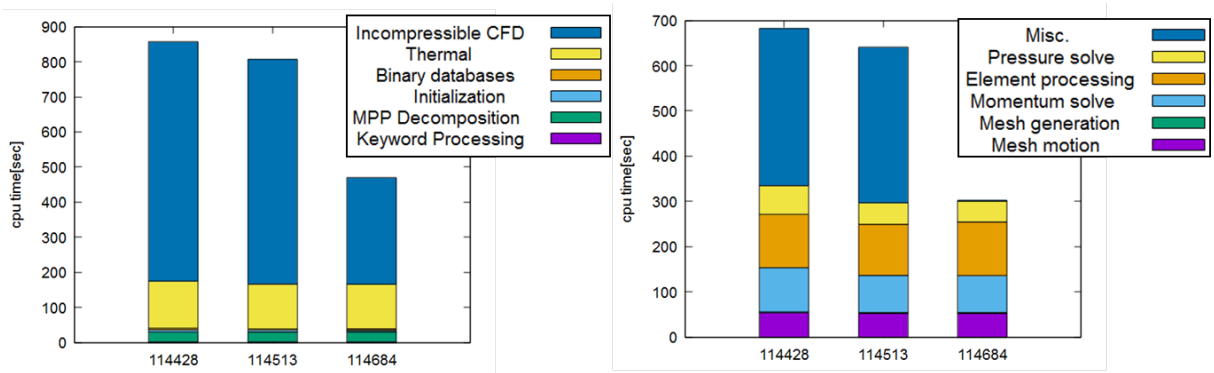

Figure 4 reduction of calculation in between revision 114428, 114513 and 114684. [Left] Cumulative graph for all computational load. [Right] Breakdown within ICFD in the left panel.

4.3 Speedup by omission of fluid mesh motion

Hot forming analysis is classified to a very particular kind of fluid-structural interaction (FSI) problem. In general FSI fluid pressure should be transferred to structure via FSI interface and deformation of the structure should change the shape of fluid domain and thus cause deformation of Eulerian mesh for the fluid. In the present case fluid and structure is coupled through their temperature and thus mesh

deformation is not necessary. Result is shown in Figure 5, One can see that a violet part named “Mesh motion” in the right figure is completely reduced. Users can turn off the mesh motion by setting *ICFD_CONTROL_MESH_MOV=-1 with development version 114684 or more recent. Of course special care should be taken to use this function since then fluid mesh does not follow the motion of the structure.

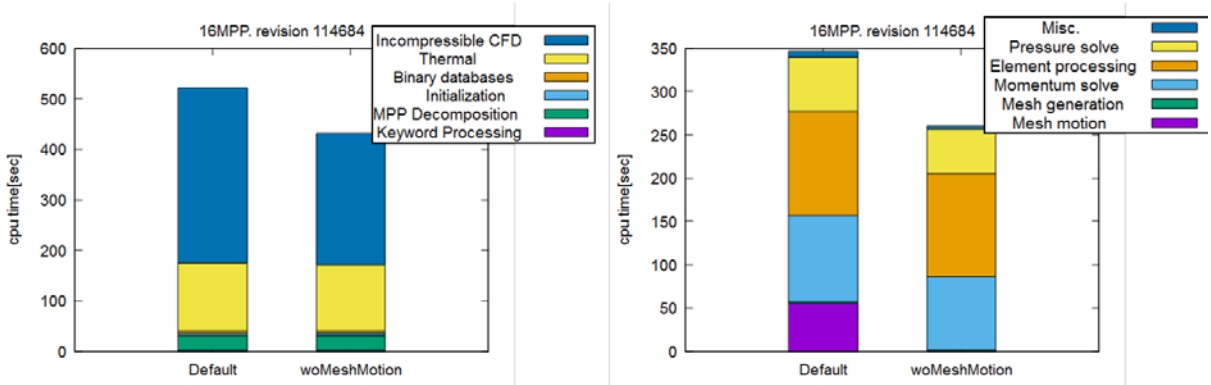

Figure 5 [Left] Comparison of calculation time for default setting and the one that mesh motion at FSI interface is disabled, available after revsion 114684. [Rgiht] Breakdown graph for ICFD load.

4.4 Speedup by shutting down Navier-Stokes calculation after the steady state

As described in section 2, timescale for fluid motion and heat conduction is well separated and thus the former equilibrates much faster than the latter in many cases. In the case with tool cooling simulation, the Plandtl number of water is about 7. It means roughly speaking time scale for fluid motion is 7 times faster than that of fluid temperature. Here we stop updating fluid motion at 1/10 of total calculation time. A special care must be taken when a user applies this function since if one stops the update before the steady state is reached the solution definitely becomes invalid. Thus note that a user has to know how long fluid motion takes to equilibrate. Ideally, by stopping the update at 1/10 of total time computational hour consumed by the fluid motion update reduce to 1/10 of original one. In addition to that, the fact that timescale for the temperature is much longer than that of the fluid motion means that one can apply larger time step after the shutdown of Navier-Stokes calculation. Actually user is not constrained by the Coulant condition of fluid motion after the shutdown. Then as a side effect of the shutdown, it is expected that the rest part of the calculation can be even reduced.

Figure 6 shows the speedup result of these two items one by one. By the shutdown of Navier-Stokes about 30% speedup is achieved and by adopting larger timestep after the shutdown one has even twice faster. The breakdown chart in the right panel tells us that “Pressure solve” and “Momentum solve” become 1/6 of original one. Since time consumed by the initialization of fluid field variables is not negligible in this calculation not as much as 1/10 reduction is achieved here. This example is only 100 steps calculation thus it is affected by the initialization procedure but for longer calculation we confirmed the ideal speedup.

By enlarging thermal timestep for the latter part of the calculation, time spent by “Thermal” and “Incompressible CFD” in the total cumulative graph become about half of original one. In the breakdown graph, “Element processing” part is reduced since it contains matrix construction for fluid temperature. We employed 10 times larger timestep after the shutdown and thus 100 steps calculation in total is reduced to 20 steps. As in the above, we confirmed that an ideal speedup is achieved if the total timestep is sufficiently large so that effect of the initialization is negligible.

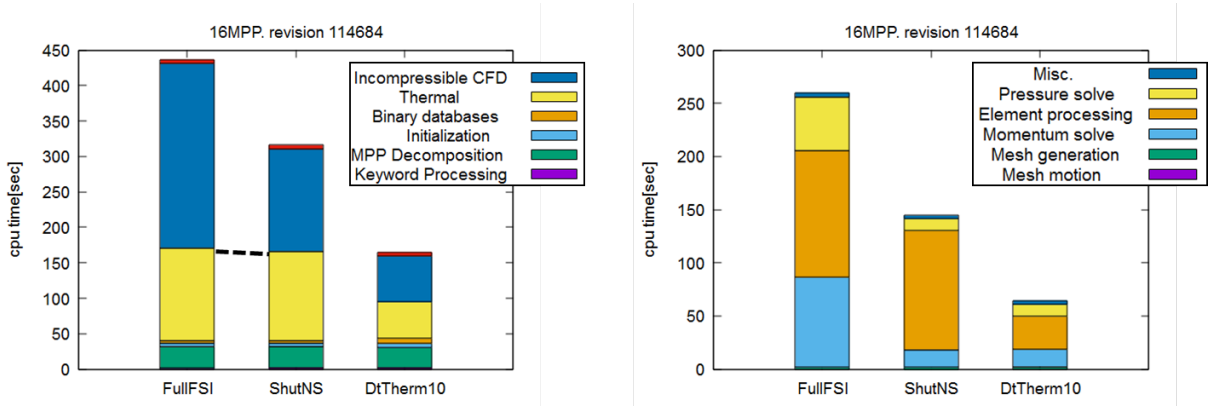

Figure 6 Speedup due to ommision of update of fluid pressure and fluid velocity is shown by „Full FSI “ and „ShutNS“. Further speedup is obtained by adopting 10 times larger timestep after shutting down Navier-Stokes calculation.

The feature to shutdown Navier-Stokes calculation can be turned on by indicating time to stop the update with the first variable of the second card of *ICFD_CONTROL_TIME. For the timestep after the shutdown, one just has to set the larger timestep by *CONTROL_THERMAL_TIMESTEP, since before the shutdown the smaller of fluid or thermal timestep is automatically employed by the solver.

5 Summery

A substantial speedup of conjugate heat transfer calculation is realized by the recent development of LS-DYNA by LSTC, based on requests and feedback from JSOL. The speedup items presented here are, use of GMRES solver, achievement of scalability by avoiding redundant calculation, and stop updating fluid motion after its steady state. In this report we did not say anything about the validity of the calculation, which is left for our future work.