Normalization of correction equation residuals

M. Syamlal, S. Benyahia, 8-22-2005

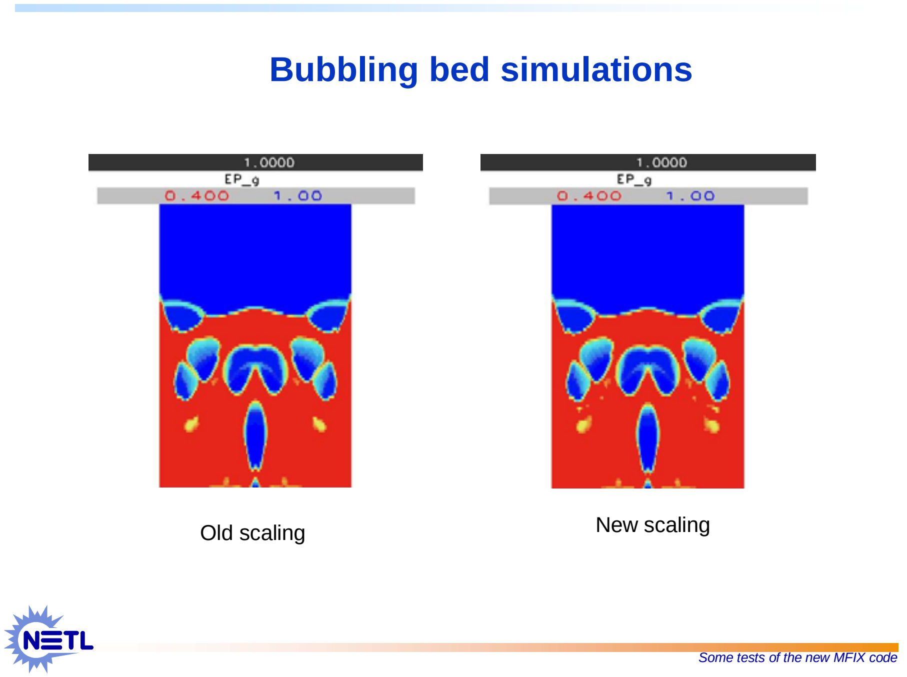
A new method for checking the convergence of pressure and solids volume fraction correction equations in MFIX was developed. The existing method uses the residual of the first iteration to normalize the residuals of subsequent iterations. The normalized residual is then compared against a tolerance value to determine whether the iterations have converged. This method at times makes the convergence criteria needlessly strict and potentially unachievable. Especially when the simulations reach a steady-state, the first iteration residual is already very small, and the use of that value for normalization, in some cases, may amount to demanding a precision exceeding the machine precision. In those instances MFIX will declare non-convergence, in spite of having a converged solution. Another option in MFIX allows the user to specify a normalization factor. This option avoids the danger of making the convergence criteria unachievable. But the onus of determining a suitable normalization factor is on the user; a poor choice of the factor may cause unacceptable errors in the solution. Furthermore, using a single fixed value for the normalization factor may not be sufficient in a transient simulation where the flow rates (boundary conditions) may change and a phase may get created (or consumed) because of chemical reactions (e.g., burning coal). These problems are avoided in the new method that uses as normalization factor one of the several terms (accumulation, fluxes, source) in the discretized continuity equation (which is the right-hand side of the pressure correction equation.) The term with the maximum value, or the most significant term, is selected as the normalization factor. This method will reduce to the standard normalization method (using the first iteration residual) for steady-state simulations with no phase changes.

The method was implemented in MFIX and tested with three short simulations. There was considerable speed up for a constant flux, periodic case (190%), which was close to a steady-state. In two transient cases the speed up was moderate (spouted bed combustor – 25%) to small (bubbling bed with a passive scalar – 6%). A comparison of the new and old results using animate_mfix shows that the new method gives results identical to the old one.

Three cases were run to demonstrate the effect of the new numerical algorithm: 1) a 2-D fully developed gas/solids flow in a channel; 2) a bubbling fluidized bed with a central jet; 3) the entrance region of a riser. The following table shows the CPU time from three detailed simulations, which all show moderate to considerable speed up.

	Bubbling bed (1 sec simulation)2	**Fully dev. Flow (60 sec simulation)**	**Riser flow (1 sec simulation)**
Old code CPU time (sec)	**53200**	**26387**	**128079**
New residual scaling CPU time (sec)	**51922** **2.5% faster**	**13528** **95% faster**	**23127** **450% faster**

The following figures show that the results with new scaling technique are identical to those with the old technique. So the considerable speed up is with no loss of accuracy.

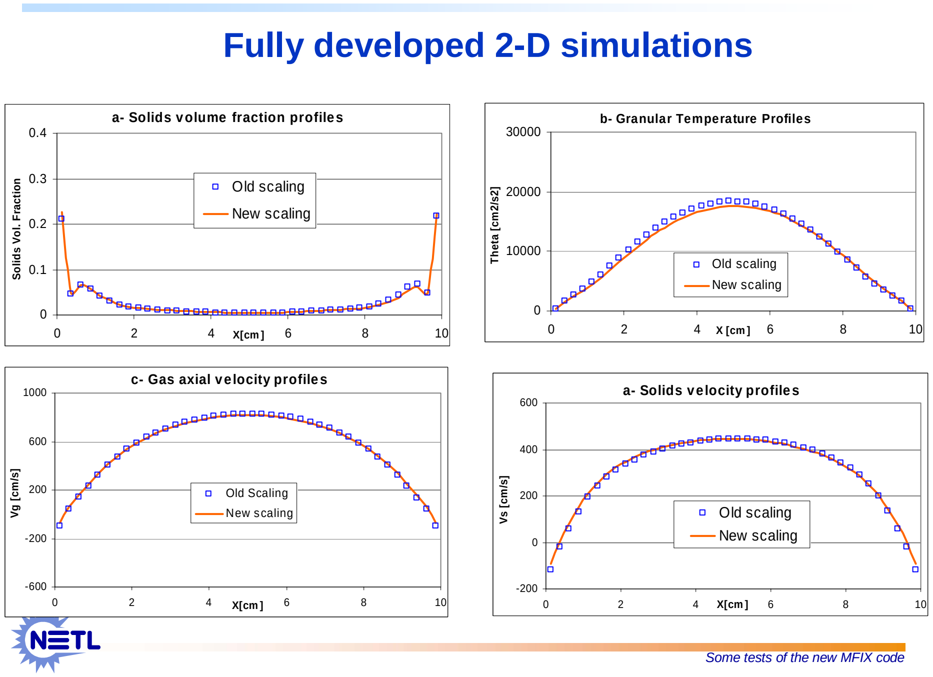


Bubbling bed simulations

Old scaling

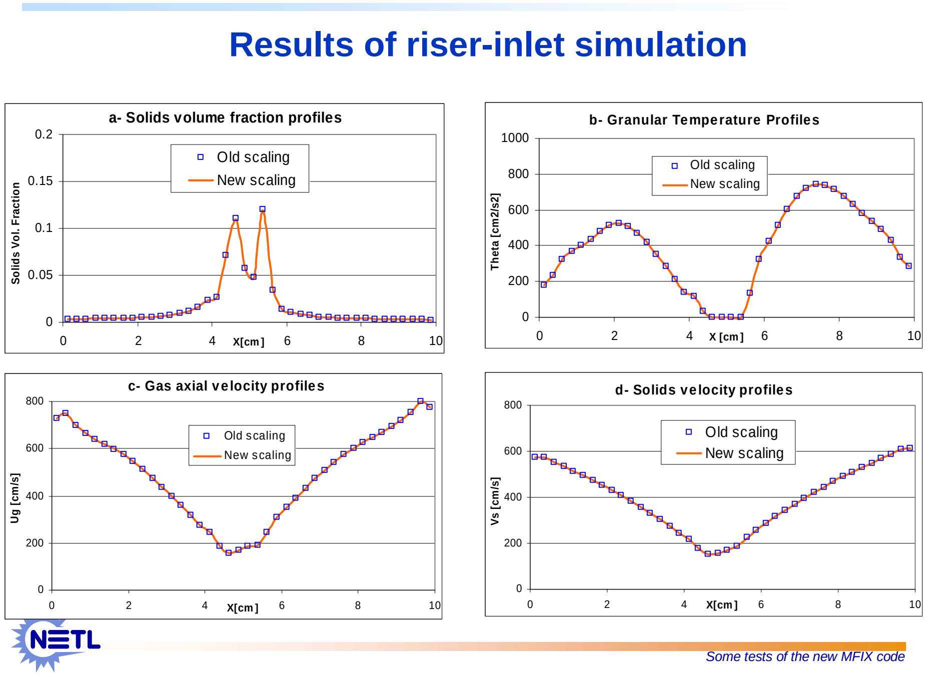
New scaling

Some tests of the new MFIX code



Fully developed 2-D simulations

Some tests of the new MFIX code



Results of riser-inlet simulation

Some tests of the new MFIX code

A parameter DEN has been defined in solve_epp.f and solve_pp_g.f to make tolerance for residual scaled with max value compatible with the residual scaled with first iteration residual. The value of DEN may be increased to tighten the convergence or decreased to loosen the convergence. It is not expected that users will need to change the value of DEN.

The new normalization method can be activated by setting Norm_g = 0. and Norm_s = 0. in *mfix.dat*.