2015 Questions of the Month

****
The January Question of the Month
****

It's been a long time since we heard from our friend Roshka, "your" employee in the computational engineering company BATALA
(Benchmarks And Theoretical Analysis for Low-tech Applications). Here is a fresh dialog between Roshka and You:

You: Roshka, welcome back. How was your vacation?

Roshka: Well, I was in Palma for 10 days, but is so happened that there was a conference there on scientific computing, so I sat all the time there, listening to lectures.

You: That's excellent, Roshka!

Roshka: Yes, but it means that I'll need a vacation soon.

You: We'll see about that. Now you are going to solve this incompressible flow problem that I gave you before you left, right?

Roshka: Yes. I have already started. I am using our code AIRS for that.

You: AIRS? AIRS is for compressible flow, Roshka. You need to use the code WATERS which is for incompressible flow.

Roshka: But I don't like WATERS, because of its inferior user-interface. I like AIRS much better, with its excellent built-in graphics. So I am using AIRS, and in order to simulate incompressible flow I set the Mach number to a very small value: 0.0001. This should give us practically an incompressible flow behavior.

You: Oy, Roshka, Roshka, what will be the end of you?

Explain your reaction to Roshka's plan.

****
Answer
****

Traditionally compressible and incompressible flow problems are handled by two separate codes, since these two types of problems typically involve different concerns and numerical difficulties. Incompressibility of the flow poses a constraint, i.e., divergence free velocity field. If we do what Roshka wants to do, namely we use such parameters that make the flow _almost_ incompressible, we are trying to _almost_ enforce the divergence-free constraint. Doing this is a naive way, using a standard compressible flow code and a reasonably crude discretization, would result in severe numerical difficulties, and the solution would be wrong. (For those of us who are more knowledgeable in solid mechanics, a similar situation occurs when we try to simulate almost-incompressible elasticity by taking the Poisson ratio to be 0.49999)

Our readers sent some additional interesting comments. RP points out the wrong results that will be obtained, as explained above. OL writes "It seems like a bad idea to solve an incompressible flow problem with a compressible flow solver with a small Mach number, since the incompressible problem has no sound waves (characteristics), whereas a compressible flow might introduce spurious sound waves." ET writes: "Using a compressible code to solve for low Mach numbers frequently has 2 problems. (1) Because of the stiffness of the ration of the highest speed u+c to the convective speed u (i.e. 1/M), any explicit code will take a long time to run especially for a steady state calculation, since the time step is governed by the fastest wave. (2) Because of the same poor conditioning, most methods, both explicit and implicit, will suffer from poor accuracy. This is caused by the upwinding or artificial viscosity which usually does not match the correct asymptotics. The resolution of these problems is (1) Preconditioning to remove the stiffness, various types can solve one of the problems or both of them; (2) Use a modified compressible system that removes the fast waves from the system."

GL and AG refer to the same difficulty mentioned above, of having to use very small time steps. AG remarks that because of this slowness of solution "Roshka can leave his computer running on the problem while he is on his vacation and after he is back it will still be running." GL adds that it would be a nice feature to have one code which could be used for both compressible and incompressible flows. Indeed this is possible, and GL cites a paper by Hirt, Amsden, and Cook (JCP, 135,(1997)) on this. There has been an earlier work on this by Hauke and Hughes (CMAME, 113, 1994).

ZZ related to the code names AIRS and WATERS, and thus assumed that AIRS is intended for inviscid compressible flow, while WATERS is intended for viscous incompressible flow. Therefore, using AIRS by Roshka would be completely wrong since it would ignore viscosity effects. AY writes: "In compressible fluid dynamics the density is an extra dynamical variable which one can discard conveniently if one assumes incompressibility from the start. Moreover, Mach's number is just the ratio between the flow velocity and sound velocity. Hence for small

flow velocities one can obtain small sound velocity which is the square root of the derivative of the pressure with respect to density (for constant entropy). Thus the derivative of density with respect to pressure is high. This means that contrary to Roshka's assumptions a small change in pressure will result in a high change in density."

Correct answers were obtained from:

Arie Gershengoren, Oren Livne, Gabi Luttwak, Ronen Payevsky, Eli Turkel, Asher Yahalom, Zvi Zaphir.

****
Remark on the January Question of the Month
****

The January Question was about Roshka's folly in trying to solve an incompressible flow problem by using a compressible-flow code. Gabi Luttwak's comment was mentioned, that it would be a nice feature to have one code which could be used for both compressible and incompressible flows. Indeed this is possible, and GL cited a paper by Hirt, Amsden, and Cook (JCP, 135,(1997)) on this. I added that there had been an earlier work on this by Hauke and Hughes (CMAME, 113, 1994). Now GL comments that an even much earlier work on this was done by F.H. Harlow and A.A. Amsden (JCP, 8, 1971), in their paper "A Numerical Fluid Dynamics Method for All Flow Speeds."

****
The February Question of the Month
****

Here is a question in the spirit of Purim, which is coming soon.

Explain the saying: MOR is less and LES is more.

If you can explain only half of this saying, this will still be regarded as a correct answer.

****
Answer
****

MOR is the acronym of Model Order Reduction (also called ROM = Reduced Order Modeling). This is a class of methods in which a discrete model containing a large number of Degrees of

Freedom (DOFs) is replaced by a model that has a much smaller number of DOFs without affecting significantly the accuracy of the model (is some desired respect). In other words, a model which requires a huge computational effort (sometimes prohibitive) is replaced by a model requiring a reasonable computational effort, without the consequence of reduced accuracy. So "MOR is less" means that Model Order Reduction requires less computing time, or less computational effort, than the standard modeling.

LES is the acronym of Large Eddy Simulation. This is a class of methods in CFD which represent large scale eddys and use some simplified model to represent or dissipate the small scales. This is in contrast to the more standard RANS in which all flow features at all scales are represented by a simplified model and by averaging. In the modeling of turbulent flows there are three levels of representation of the physics: RANS (the most standard and most efficient, using a "turbulent model" from the outset), LES, and DNS (Direct Numerical Simulation) in which the Navier Stokes equations are solved fully and directly (which requires a super-computer and typically involves millions of DOFs). So "LES is more" means that LES represents "more physics", or is more accurate with respect to reality, than the standard analysis method.

Correct answers were received from:

Michel Bercovier, Steven Frankel, Arie Gershengoren, Oren Livne, Gabi Luttwak, Ronen Payevsky, Eli Turkel

****

The March Question of the Month

****

Once again, recall that you are the boss of Roshka in the computational engineering company BATALA (Benchmarks And Theoretical Analysis for Low-tech Applications). Here is a dialog between Roshka and you.

Roshka: Boss, I came to tell you that I am leaving in one month. I received an offer from our competitive company NIRVANA (Numerical Investigation, Results Verification and Analysis for Naive Applications), and accepted it.

You: Well, Roshka, what can I say? I wish you success. NIRVANA is certainly a good company, although I hear that their employees are not as relaxed as ours.

Roshka: Of course I will finish the linear axi-symmetric simulation that you asked me to do.

You: Yes, how is it going with this simulation?

Roshka: You know, something strange is happening. I am using the linear commercial software we always use. I defined the geometrical model, giving the (x,y) coordinates of all the keypoints. Then I generated the mesh. I indicated to the software that this is a 2D simulation, and then I indicated that this is an axi-symmetric simulation (and not a plane-problem simulation). The mesh is very nice and dense, with 40,000 DOFs, much more than we really need for capturing the phsyics, and with well-proportioned and nicely shaped elements. The model ran without any indication of a mistake or a problem. The results have a very reasonable order of magnitude. However, the results cannot be right because they are completely different (not just a little different) than the analytic solution that we have for a similar problem, than our lab experiments, and than other numerical results that I have found in the literature. I don't have a clue what could be wrong.

You: Here is the first thing that you have to check, Roshka.

What is this first thing that Roshka has to check?

****
Answer
****

As some readers commented, many mistakes are possible, e.g., in the boundary conditions, initial conditions, material properties... However, what I had in mind (and most readers have suggested this) is to check whether the axis of symmetry as Roshka meant it is indeed the axis of symmetry that the software assumes. In a 2D model, with the coordinate system (x,y), the axis of symmetry may be either the x axis or the y axis. Usually it is assumed to be the y axis, since the system of coordinates is actually a cylindrical one, (r,z), where z is the axis of symmetry, and so x=r and y=z. (But MB comments that in some codes the axis of symmetry is the horizontal axis.) In any case, it may have happened that in Roshka's model the axis of symmetry was meant to be the x axis, while the software assumed that it was the y axis. Such a mistake would lead to a result that looks completely innocent, with numbers that have the correct order of magnitude, but is of course completely wrong.

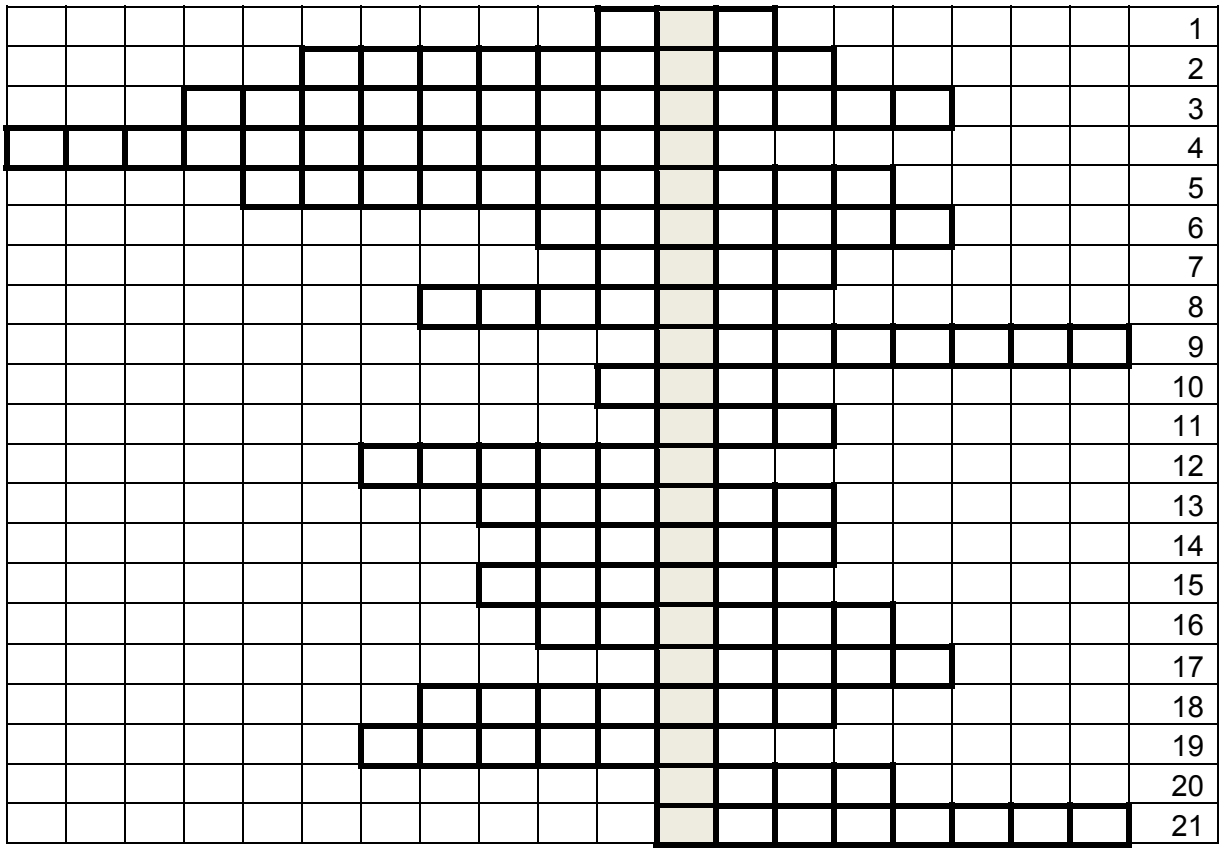
Correct answers were received from:

Orna Agmon Ben-Yehuda, Hanan Amar, Rami Ben-Zvi, Michel Bercovier (look), Nir Emuna, Meir Feder, Amiel Herszage, Oren Livne, Gabi Luttwak, Zvi Zaphir

****

The April Question of the Month

****



הגדרות:

1. ציוד של דייג נומריקאי
2. בדיקה אם אנו פותרים את המשוואות נכון
3. שיטה ידועה המבוססת על עקרון מינימום או על צורה חלשה
4. שיטה ידועה המבוססת על קירוב של נגזרות
5. שיטה ידועה וגם מעצבי מתכת מוסלמים-מיסטיים
6. "סיבוב" אחד של שיטה שחוזר על עצמו
7. ההתפשטות שלהם מיוצגת ע"י אחת מהמשוואות הלינאריות החשובות
8. שיטה חשובה שהומצאה ע"י הישראלי אכי ברנדט
9. בדיקה שאנו פותרים את המשוואות הנכונות
10. הזרימה שלו מיוצגת ע"י אחת מהמשוואות הלינאריות החשובות
11. בעיה לא לינארית במכניקת המוצקים וגם סוג קרב
12. מטריצה חשובה וגם תכונה חשובה למפקד טירונות
13. (חישוב) סימולטני, טוענים שנשים טובות בו יותר מגברים
14. מדען, לפעמים קדמי ולפעמים אחורי
15. מצב שבו הפתרון כבר לא משתנה, וגם מי שלא מפסיק את פעילותו

16. סימן חדש: משפחת שיטות צעידה בזמן
17. מה שמקבלים כאשר מציבים פתרון לא מתאים למשוואה הדיסקרטית
18. לפני כל אסיפה: התקרבות לפתרון המדויק
19. כשהתפוח נפל הוא גילה שיטה לפתרון בעיה אלגברית לא לינארית
20. חבר של זיידל שפיתח שיטת אינטגרציה נומרית
21. סוג שיטה המתאפיינת בהתכנסות מהירה במיוחד

What you need to do is to write the words (in Hebrew) defined below the "table" in the corresponding thick cells. If you do this correctly, you will be able to read a theorem in the vertical shaded column. Your answer should be just this theorem.

****

Answer

****

The solution is written in the “table” below.

										ת	ש	ר						
					ה	י	צ	ק	י	פ	י	ר	ו					
			ם	י	י	פ	ו	ס	ם	י	ט	נ	מ	ל	א			
ם	י	י	פ	ו	ס	ם	י	ש	ר	פ	ה							
				ם	י	י	פ	ו	ס	ם	י	ח	פ	נ				
									ה	י	צ	ר	ט	י	א			
										ם	י	ל	ג					
							ג	י	ר	ס	ב	ר						
											ה	י	צ	ד	י	ל	א	ו
										ם	ו	ח						
											ע	ג	מ					
						ת	ו	ח	י	ש	ק							
								י	ל	י	ב	ק	מ					
									ר	ל	י	ו	א					
								ד	י	מ	ת	מ						
									ק	ר	מ	ו	י	נ				
											ת	י	ר	א	ש			
							ת	ו	ס	נ	כ	ת	ה					
						ן	ו	ט	ו	י	נ							
											ס	ו	א	ג				
											ת	י	ל	ר	ט	ק	פ	ס

The theorem that can be read vertically is (translated into English): "A method which is stable and consistent is convergent." This is the famous theorem of Peter Lax (http://en.wikipedia.org/wiki/Peter_Lax).

Comment: I made a spelling mistake in Achi Brandt's name. It should be אחי. Sorry about this. See http://en.wikipedia.org/wiki/Achi_Brandt and http://he.wikipedia.org/wiki/%D7%90%D7%97%D7%99_%D7%91%D7%A8%D7%A0%D7%93%D7%98 .

Correct answers were received from:

Rami Ben-Zvi, Anne Weill.

****

The May (and last for a while) Question of the Month

****

Here are 10 statements made by our friend Roshka. Some of them are correct and some are not. Point to at least 3 statement that are wrong, and explain why they are wrong.

1. Solution by Gauss elimination of a linear algebraic system obtained from the Boundary Element Method (BEM) discretization with N degrees of freedom, requires O(N^3) operations.

2. Problems of fluid-solid interaction must be solved iteratively, where one solves alternately the solid problem and the fluid problem, until convergence is achieved.

3. There is no linear multistep time-stepping method which is explicit and unconditionally stable.

4. In wave problems with a fixed frequency, one has to take between 10 and 20 grid points per wave length.

5. Contact problems where the contact region is not known a priori are always nonlinear problems.

6. In problems involving a viscoelastic material, it is required to store and use, in every time step, the entire solution history in all the previous time steps.

7. In a spectral or high-order method based on a grid/mesh, uniform spacing between the grid points will not give good results.

8. Using a compressible flow code, it is easy to solve also incompressible flow problems.

9. In order to obtain the steady-state solution, one can solve the time-dependent problem in an efficient way, which does not provide accuracy in time, but reaches an accurate approximation to the steady state.

10. In parallel computation, the computing time is reduced linearly with the increase in the number of processors.

****
Answer
****

Here are the 10 statements again, with some comments on their truthfulness.

1. Solution by Gauss elimination of a linear algebraic system obtained from the Boundary Element Method (BEM) discretization with N degrees of freedom, requires O(N^3) operations.

This was meant to be a trap. Finite Difference, Finite Elements and Finite Volumes methods produce algebraic systems with sparse matrices, namely matrices that contain only a small number of nonzero entries. With such matrices, the number of operations is much smaller than O(N^3). However, the BEM is different. On one hand its discretization is one dimension lower than that of the methods mentioned above. For example, if the original problem is 3D, only the 2D boundary is discretized in BEM. Therefore the matrices that BEM produces are generally much smaller than those produced by the other methods. However, these BEM matrices are full, not sparse. Therefore, indeed a BEM linear system of equations with an NxN matrix requires O(N^3) operations. So this statement is generally **true**.

In some special cases, there are sophisticated methods to solve the BEM equations (e.g., the multipole method of Greengard) which require a much smaller number of operations, typically O(N logN).

2. Problems of fluid-solid interaction must be solved iteratively, where one solves alternately the solid problem and the fluid problem, until convergence is achieved.

False. There are also "monolithic methods" which solve the fluid and solid problems simultaneously, as one big system of equations and interface conditions. GL mentions the scheme of W. Noh. We have had a Question of the Month dedicated to this issue.

3. There is no linear multistep time-stepping method which is explicit and unconditionally stable.

True. There are a bunch of beautiful theorems proved by the late Swedish mathematician Dahlquist, called the "Dahlquist Barriers", which relate to a certain wide class of time-stepping methods, called linear multistep (LMS) methods. These theorems tell you what you _cannot_ achieve by LMS methods. Achieving an explicit unconditionally-stable method is impossible, and this is one of the barriers. See, e.g., Section 5 in https://en.wikipedia.org/wiki/Linear_multistep_method#First_and_second_Dahlquist_barriers . By the way, time-stepping methods which are explicit and unconditionally-stable _do_ exist, but they are not LMS methods, in that they use a time-step that is not constant but depends on the solution itself in past times.

4. In wave problems with a fixed frequency, one has to take between 10 and 20 grid points per wave length.

This is a popular "rule of thumb" but it is actually **false.** We have dedicated to this one the of the past Questions of the Month. The number of grid points per wave length that have to be taken in order to maintain a certain accuracy is not constant. There is no such "magic number". This number actually depends on the wave length. For long waves (low frequencies), the "rule of thumb" above may be appropriate, but for shorter waves (higher frequencies) one would need a higher number of grid points per wave length. Early theorems on this issue have been proved by Eli Turkel, and later Babuska and others returned to this issue and proved additional things.

5. Contact problems where the contact region is not known a priori are always nonlinear problems.

True. The contact region has to be parametrized in some way; for example one may distinguish between nodes that are "free" and nodes that are "in contact". Identifying the contact nodes at each time or for each load value is part of the problem and the equations that governs this identification are highly nonlinear. There are various mathematical models for contact, with or without friction. One type of model is based on a (convex) quadratic programming problem.

6. In problems involving a viscoelastic material, it is required to store and use, in every time step, the entire solution history in all the previous time steps.

In principle this is true, since viscoelastic materials have "memory". However, in most cases one would represent the viscoelastic material kernel (or would approximate it) by a Prony series, namely a sum of exponential functions, and in this case it is possible to manipulate the convolution equation so as to obtain a scheme that does not require memory. Many years ago Rami Ben-Zvi wrote a paper about this trick. If one uses this popular trick then the statement is **false**.

7. In a spectral or high-order method based on a grid/mesh, uniform spacing between the grid points will not give good results.

This is generally **true,** except maybe for some very special methods**.** We dedicated a Question of the Month to this issue. A spectral scheme of high order with a uniform grid suffers from instability, which manifests itself as the Runge phenomenon. See, e.g., https://en.wikipedia.org/?title=Runge%27s_phenomenon . For this reason, special grid spacings have been invented which prevent such instability, like Chebyshev (see, e.g., https://en.wikipedia.org/wiki/Chebyshev_nodes) and Legendre spacings.

8. Using a compressible flow code, it is easy to solve also incompressible flow problems.

False. We have dedicated a Question of the Month to this issue not long ago. An incompressible flow problem requires special computational techniques to enforce the incompressibility constraint in an appropriate manner. Simply taking very small artificial incompressibility would produce bad results. I will not repeat here the interesting explanations provided by some readers on this question.

9. In order to obtain the steady-state solution, one can solve the time-dependent problem in an efficient way, which does not provide accuracy in time, but reaches an accurate approximation to the steady state.

True. This one too was discussed in a previous Question of the Month. There are actually two approaches to solve nonlinear steady-state problems: either by iterations on the elliptic problem (e.g., Newton iterations), or by time-stepping in the manner described in the question ("pseudo-time methods"). In the past the latter approach dominated CFD, but in recent years the former gains more and more popularity, and as Michael Engelman has told us, many software packages now prefer the former approach.

OL comments on some possible pitfalls of the pseudo-time approach: "What if there is no steady state solution so we can't converge to anything? Or it exists but unstable, so every little perturbation to a wavelength's amplitude in the time dependent problem will create a large error in the steady state limit."

10. In parallel computation, the computing time is reduced linearly with the increase in the number of processors.

False. This would have been true if the algorithm was perfectly parallelized and communication between the processors was infinitely fast. In reality, communication takes time. In fact, if the algorithm or the implementation is not suitable for parallelization, the communication time may dominate, in which case increasing the number of processors would hardly lead to any speed-up in the solution process. GL also remarks that in any algorithm there are parts which cannot be parallelized. Rarely, an algorithm exhibits almost linear speed-up with the increase in the number of processors, and then it is said to be a "ridiculously scalable" algorithm.

Correct answers were received from:

Oren Livne, Gabi Luttwak.

Tam Velo Nishlam...