

流体シミュレーション

イメージだけでは…

一杯の紅茶を淹れる時のことを考えてみてください。

あなたは電気ケトルで沸かしたお湯をお茶葉の入ったティーポットに注ぎ、さらにそこから紅茶をティーカップに注ぐ。この一連の流れをイメージするのは簡単なことだと思います。
ではこの間に、一体どのくらいの水滴がテーブルに飛び散ったのでしょうか。

もうちょっとぐっとイメージを膨らませてみましょう。電気ケトルから注ぐときのお湯は、グツグツと沸き立っていてたくさんの水滴が飛び散るでしょうし、高い位置から注いだ場合の方がより多く飛び散りそうですね。

ですがもっと正確に、どのタイミングでどこに何滴落ちたかとなるとどうでしょう。さすがにイメージでは難しくなってきますよね。そういったことを計算で正確に求めるために用いるのが、本作品で扱っている流体シミュレーションです。
次の式を見て下さい。

ナビエ-ストークス方程式

$$\frac{\partial v}{\partial t} + (v \cdot \nabla) v = -\frac{1}{\rho} \nabla p + \nu \nabla^2 v$$

何のことだかよくわからないかもしれませんが、これが流体シミュレーションで最も重要な方程式なのです。この世の水や空気は流れ行くままに動いているかのように見えて、実はこのナビエ-ストークス方程式にピッタリとはまって動いているのです。むろん先ほどの紅茶のお話も例に漏れずこの方程式は、水滴が、いつ、どこに、何滴落ちるのかを正確に言い当てているのです。

では、これからこの方程式が具体的に何を言っているのかを、そしてコンピュータ上でどのように計算するのかをみていきましょう。

ナビエ-ストークス方程式とは...

ナビエ-ストークス方程式は単純に言えば、

水のごく小さな部分、すなわち大きな流れを一滴一滴区別してみたときの「速度」と「圧力」がわかれば、次の瞬間の「速度」と「圧力」もわかる。その一滴一滴の動きをあわせたものが流れなのだ。

ということを言っているのです。

いま、次の瞬間と言いましたが、次の瞬間の流れが分かればまた次の瞬間そしてまた次の瞬間というようにずっと先の流れまでわかることになるのです。

式の中では、 v が速度 t が時間 p が圧力 に対応しています。

より詳しく説明していきましょう

左側の $\frac{\partial v}{\partial t} + (v \cdot \nabla) v$ という式。

これが、「現在の速度と次の一瞬の速度がどれくらい違うのか」ということを表している部分です。その違いが大きい時もあれば小さい時もある。それがどのくらいの大きさなのかを残りの式が教えてくれます。

残りの式とは $-\frac{1}{\rho} \nabla p$

と $\nu \nabla^2 v$ のことです。

これらはともに、一滴に加わる力を表しています。一つ目は、圧力によってぐぐっと「押される力」。二つ目は粘り気によって生まれる「引きずられる力」です。「引きずられる力」というのが、ちょっとイメージしづらいですね。目に見える現象で言うならば、水が斜面を流れているとき、水はさらさらしていますから「引きずられる力」はほとんどありません。ですが、この斜面をドロドロの水飴が流れている場合だとどうでしょう。なかなか進めずに、ゆっくりと下っている様子が想像できますよね。これは地面からの「引きずられる力」による物なのです。この力は、流体と地面の間だけではなく流体同士、つまり隣合う一滴一滴の間にも働くのです。ちなみに、これらの力は全て現在の時刻のものです。以上まとめると、ナビエ-ストークス方程式は

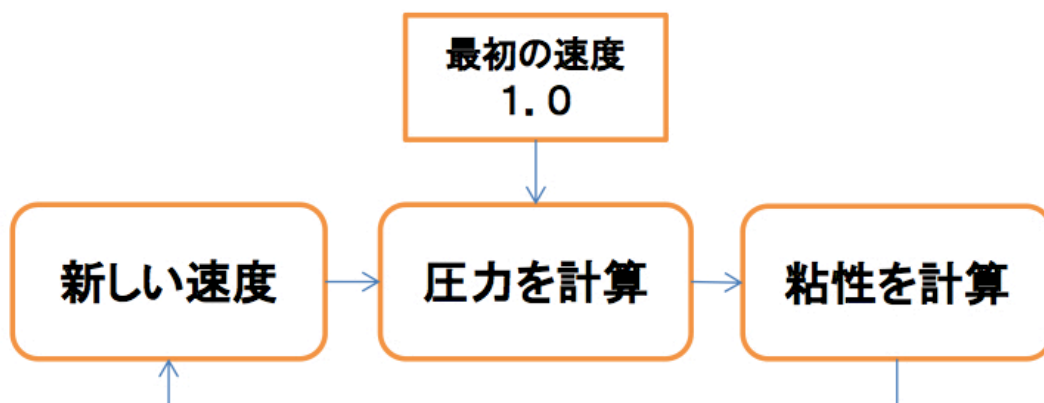
$$(\text{一瞬後の速度}) - (\text{現在の速度}) = (\text{現在の圧力による力}) + (\text{現在の粘り気による力})$$

ということになります。圧力や粘り気といった多様な「力」が、次から次へと「新たな流れ」を作り出しているといったところでしょうか。

コンピュータに命令をする

さて、ここからこの方程式ををコンピュータにわかるように書いてコンピュータに指示を出さなければなりません。「コーディング」と呼ばれる作業です。このシミュレーションの「コーディング」を順に説明していきますね。

いま考えているのは、左から右への一定速度の流れの中に柱が一本立っている様子を上から見たものです。まず、長方形を 200×400 のマスの区切ります。この区切った一つ一つのなかで速度はどうなっているのか、圧力はどうなっているか、時間ごとに考えていくわけです。一番最初の段階では、全てのマスで速度が 1.0、圧力はきまっていないという状態からのスタートです。そこから現在の圧力を求め、粘り気による力を求め、新しい速度を作る。またそこから圧力を..... この繰り返しです。図にすると...



といった感じです。図にすると単純ですが、矢印が一つ分進むごとに膨大な量の計算が行われています。一つ一つの計算がすんなりと終わるものではないということに加え、この計算を 200×400 個の全てのマスに関して行うわけですからね。

さて。計算の大まかな流れは上に書いた通りなのですが、これだけでは左から右に、一定の速度で水が流れているだけのシミュレーションになってしまいます。今回考えているのはその流れの中に柱が一本立っているというものでしたね。これによって流れが様々に変化していくのです。

そのために境界条件というものを、計算に加えます。

どういうことかということ、柱の中に水は流れていませんから速度はもちろん0です。では、柱の周囲はどうでしょう？柱があるせいで、普通だったらそのまま進めるはずだった場所で渋滞が起こって速度が落ちていくでしょう。柱のすぐ外側では、水はもはや進むことができませんから速度は0になるでしょう。そこでうまく速度を調整して、柱のすぐ外側で0になるようにしようということなんです。

コンピュータへの命令はこんなところですね。

実際のアニメーション

ここまで書いたようなことを実際に行っているのがこのアニメーションです。なんとなく水が流れているように見えるでしょうか？（冒頭に書いたような、一滴一滴を正確に予言するようなプログラムにはほど遠いのですが...）

YouTubeで「物理シミュレーション」と検索してみると、流体シミュレーションだけでなく色んな面白いCGアニメーションがあるので興味がある方は見てみると面白いですよ。