INF1608 – **Análise Numérica Trabalho Final: Movimento de um Pêndulo**

Prof. Waldemar Celes Departamento de Informática, PUC-Rio

Autores: **Rafael AZEVEDO (1221020); Rodrigo PUMAR (1221007)**

1. Introdução

As simulações físicas são uma das motivações para implementações de métodos numéricos, às vezes por não possuírem solução analítica mas visando principalmente simular situações complexas por métodos numéricos computacionalmente rápidos (ou aceitáveis).

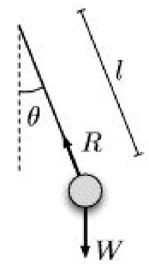
O objetivo do trabalho desenvolvido consiste em analisar a dinâmica do sistema mecânico que constitui um pêndulo simples composto por uma barra rígida de massa desprezível e comprimento L com um corpo puntiforme de massa m em sua extremidade. As forças dissipativas não são consideradas.

O problema do pêndulo simples é um problema que requer a solução de equações diferenciais não-lineares. Entretanto, uma versão simplificada da fórmula do pêndulo pode ser utilizada para ângulos relativamente pequenos. A simulação pretende ainda identificar o maior ângulo para o qual a solução das equações do pêndulo simples permanece aproximadamente correta. A medida para avaliar a validade da solução obtida pela modelagem matemática simplificada do problema do pêndulo simples foi a equação de T indicada abaixo (a avaliação será feita com base no cálculo do período do pêndulo). A versão simplificada do problema do pêndulo simples utiliza a equação diferencial indicada abaixo, aproximando $\sin(\theta)$ por θ.

Caso θ seja pequeno suficiente, espera-se obter uma solução pelo método de Runge Kutta de 4ª ordem que indique um período igual ao calculado pela fórmula padrão do período T para o caso da simplificação.

A solução analítica para grandes ângulos pode ser auxiliada por métodos numéricos para evitar cálculos complexos e resolver a equação diferencial em tempo pelo menos próximo do tempo real, permitindo simular o movimento do pêndulo. Portanto, o trabalho desenvolvido aplica o método de Runge Kutta de 4ª ordem para simular o pêndulo em diversos ângulos iniciais.

A comparação da solução para pequenos ângulos de abertura (iniciais) com a solução para os mesmos ângulos para o modelo de pêndulo simples simplificado será objeto de estudo, assim como a avaliação da discrepância entre os resultados obtidos com o modelo analitico e com os métodos numéricos para grandes ângulos θ de abertura (iniciais, com velocidade possivelmente nula).



A equação diferencial de segunda ordem que rege o movimento do pêndulo está indicada abaixo.

$$\ddot{\theta} + \frac{g}{l}\sin\theta = 0$$

Compilação: gcc -o main main.c simulacao.c taylor.c -lm

2. Desenvolvimento

a. Análise do Problema

O problema possui parâmetros iniciais para a simulação, nomeadamente o ângulo inicial (θ), o comprimento da haste (L), considerado 5 metros nas simulações realizadas, a aceleração da gravidade (g = 10 m/s) e o tamanho do passo do método numérico de Runge Kutta (h).

Sistema: d(θ) / d(t) = velocidade
d(velocidade) / d(t) = (g / l) * sin(θ)

Inicialmente, o método de Runge Kutta utiliza um passo de tamanho fixo h. No caso, o passo indica a variação do domínio do tempo para a qual o método numérico Runge Kutta é executado, determinando a solução no instante t(inicial) + h. Porém, o tamanho fixo do passo implica que o passo pode ser muito grande para alguns valores do domínio, agregando ao resultado do método numérico um erro que poderia ser minimizado caso o passo fosse menor.

Em outros casos, o passo pode ser muito pequeno desnecessariamente dado que um passo maior não agregaria pior erro ao resultado e aumentaria o tempo disponível para o passo ser calculado pelo método numérico em tempo real (tempo de cálculo do passo deve ser menor ou igual ao passo).

Logo, um passo muito pequeno aumenta o tempo total da simulação mas exige que a execução de cada passo demore cada vez menos tempo, no máximo o tempo correspondente ao valor do passo h, para que a simulação possa ser em tempo real. De fato, deseja-se que o passo seja maior ou igual ao passo. Segundo a observação dos resultados, para passos menores 10ms, a simulação em tempo real pelo método Runge Kutta adaptativo revela-se inviável.

A função de Runge Kutta de passo fixo foi implementada de 4 formas diferentes para determinar os valores da função $\theta(t)$ em um dado intervalo de simulação para o ângulo. Portanto, durante o trabalho vamos analisar a função $\theta(t)$ de 3 maneiras, considerando um passo fixo.

1) Runge Kutta de passo fixo no modelo simplificado: RKS
   $\theta(t) = -g/l * \theta$
2) Runge Kutta de passo fixo no modelo representado por serie de taylor: RKT (o valor da função $\theta(t)$ é calculado usando uma série de Taylor de grau 5 para calcular o valor de $sin(\theta)$).
   $\theta(t) = - g/l * sin(\theta)$
3) Runge Kutta de passo fixo de máxima precisão: RKMP (usando a função sin da biblioteca math.h).
   $\theta(t) = - g/l * sin(\theta)$

Adaptando o método de Runge Kutta para um passo variável (método adaptativo), podemos estabelecer uma tolerância para aceitar o passo, podendo aceitar um passo menor se o erro for menor. O passo adaptado será determinado comparando e calculando a diferença do erro entre o resultado obtido com o passo h e o resultado obtido com dois passo consecutivos de tamanho h / 2.

Se a diferença entre os resultados for menor que a tolerância estabelecida, aceitamos o passo e podemos aumentar o passo. Caso contrário, diminuímos o passo pela metade e utilizamos esse passo menor duas vezes consecutivas.

A fórmula para aumentar e diminuir o passo é a mesma para diminuir e aumentar:

erro = $\theta1(dois\ passos\ de\ h/2) - \theta2(um\ passo\ de\ h)$

$hnovo = hantigo * \sqrt{TOLERANCIA\ /\ erro}$.

A simulação com passo variável e com o modelo de máxima precisão é a última simulação realizada.

4) Runge Kutta passo adaptativo de máxima precisão: RKADAP-MP

Os erros de cada um desses 4 métodos são comparados através do valor calculado para o período, utilizado como valor de referência o valor exato da solução por série de potências de integral elíptica apresentada abaixo, truncada para no termo θ^{8} (Ref[1]).

$$T = 2\pi\sqrt{\frac{\ell}{g}}\left(1 + \frac{1}{16}\theta_0^2 + \frac{11}{3\,072}\theta_0^4 + \frac{173}{737\,280}\theta_0^6 + \frac{22\,931}{1\,321\,205\,760}\theta_0^8 + \frac{1\,319\,183}{951\,268\,147\,200}\theta_0^{10} + \frac{233\,526\,463}{2\,009\,078\,326\,886\,400}\theta_0^{12} + \cdots\right).$$

b. Desenvolvimento do Runge Kutta de 4ª Ordem

Há duas funções desenvolvidas para executar o método de Runge Kutta utilizando passo (t) de tamanho h.

- void solveRungeKutta(double h, double tet0, double w0, double g, double l, double* r, double(*f) (double ftet, double fgl), double *t);

  Essa função executa o método Runge Kutta com um passo de tamanho fixo (h). Dado um ângulo $\theta(t)$, 4 termos são calculados para determinar o resultado do passo. Os valores recebidos pela função solveRungeKutta estão descritos abaixo.
  double h - o tamanho do passo para essa iteração;
  double Tet0 - Ângulo do passo anterior (ou inicial no caso do primeiro passo);
  double w0 - velocidade do passo anterior (ou inicial no caso do primeiro passo);
  double g - Aceleração da gravidade
  double l - Comprimento da haste.

- double solveRungeKuttaAdaptativo(double h, double tet0, double w0, double g, double l, double* r, double(*f) (double ftet, double fgl),double emax, double *t, int *evol);

  Essa função utiliza a função solveRungeKutta 3 vezes, uma vez para o passo h e outras duas para o passos h/2. A partir disso, os resultados obtidos são utilizados para avaliar a diferença entre os resultados obtidos com um passo h e com dois passos h/2, considerando o valor de tolerância emax.

3. Resultados e Análise

Os resultados são impressos durante a execução do programa. De acordo com os resultados observados a partir da execução do programa de simulação desenvolvido, o modelo de Taylor [RKT] apresenta piores resultados embora tenho um tempo de simulação menor que o método com modelo simplificado [RKS], conforme o esperado. O segundo modelo de método Runge Kutta com pior resultado (baseado na diferença entre o calculado e o previsto) é o modelo simplificado [RKS].

O modelo de máxima precisão [RKMP] apresenta o melhor resultado entre os modelos de de passo fixo, porém apresenta o custo mais alto, sendo o método com o maior tempo de execução/simulação. O modelo adaptativo [RKADP-MP] reduziu o tempo de simulação consideravelmente em comparação com os métodos com passo constante bem pequeno, como por exemplo o passo h = 0.0001, revelando que esse método apresenta melhor resultado par se aproximar de uma simulação real.

Em relação a ângulos menores que 0.244 rad, o modelo simplificado apresentou erros na ordem de 10^{-3}. Porém, conforme os ângulos aumentaram, os erros aumentaram rapidamente para ordem de 10^{-1}, demonstrando que o método é válido para ângulos pequenos até 0.244 rad.

Todas as simulações tiveram o tempo de execução muito menor que 4 segundos e, portanto, menor que o período. Quando usamos um tamanho de passo inicial apropriado, em especial usando passo variável com tolerância de 10^{-8}, conseguimos executar toda simulação na faixa de 4ms, 1000 vezes mais rápido que o período total.

Durante a análise, verificamos o tempo de execução de um passo, observando quando o tempo de execução de um passo do método excede o tamanho do passo (tempo). A linha de resultado STEP TIME EXCEEDED possui apenas dois valores válidos, 1 e 0. Caso seja 1, então houve pelo menos um passo cujo tempo de execução excedeu o tamanho do passo h.

Essa análise revelou que algumas implementações do método de Runge Kutta apresentaram o tempo de execução de um passo maior que o tempo do passo em si, inviabilizando a execução da simulação deste passo em tempo real. Isso ocorreu em especial no passo adaptativo, quando o passo não foi aceito inúmeras vezes consecutivas devido ao passo inicial ser muito maior que a tolerância. Entretanto, para passos constantes (fixos) a execução de um passo exceder o tamanho do passo revelou ser algo muito frequente, principalmente para passos muito pequenos.

A discrepância do resultado da ultrapassagem do passo pelo tempo de execução e do tempo total de execução ser muito menor que o período total indica que existem passos críticos que dificultam, em alguns casos, a execução de todos os passos em tempo real.

Porém, antes que um passo crítico ocorra, os passos anteriores foram calculados em um tempo muito inferior, então o passo crítico ganha o tempo economizado nos passos anteriores que não foram executados em tempo maior que o tamanho do passo. Logo, se 100 passos consecutivos foram calculados em menos tempo que o passo dado, então o tempo acumulado do passo não gasto pelo cálculo de cada um desses passos garante em geral que um passo futuro com tempo de execução pelo método maior que o tamanho do passo tenha um tempo extra para ser determinado mantendo a simulação em tempo real. Portanto, concluímos que a simulação como um todo pode ser mostrada em tempo real com um passo adequado e uma análise para verificar que passos críticos (levam mais tempo para serem calculados que o valor do passo) são executados não excedendo o tempo em que a simulação está adiantada em relação ao tempo previsto.

Bibliografia:

[1]https://en.wikipedia.org/wiki/Pendulum_(mathematics)