数値計算の理論と実際レポート課題 3

16T1601W 杉山隼太

3 体問題についてルンゲクッタ 2 次，4 次で解き，$x-y$平面における 3 質点の軌道と，エネルギー誤差の時間変化について調べる．図 1 に作成したプログラムのソースコードを示す．

```
#include <iostream>
#include <cstdlib>
#include <cstdio>
#include <cmath>

using namespace std;

static const int NMAX = 128;

//ルンゲクッタ用の変数
static double xtmp[NMAX][3];
static double k[NMAX][3], k1[NMAX][3], k2[NMAX][3], k3[NMAX][3], k4[NMAX][3];
static double l[NMAX][3], l1[NMAX][3], l2[NMAX][3], l3[NMAX][3], l4[NMAX][3];

void gravity(double (*x)[3], double *m, double (*a)[3], double *p, const int n) {
		for (int i = 0; i < n; i++) {
				//各質点にはたらく重力を計算する
				for (int j = 0; j < 3; j++) {
						a[i][j] = 0;
				}

				for (int j = 0; j < n; j++) {
						if (i != j) {
								double dr = 0;
```

```
					for (int k = 0; k < 3; k++) {
						dr += pow(x[j][k] - x[i][k], 2);
					}
					dr = sqrt(dr);

					for (int k = 0; k < 3; k++) {
						a[i][k] += 1.0 * m[i] * (x[j][k] -
x[i][k]) / pow(dr, 3.0);
					}
				}
			}
	}
}

double energy(double (*v)[3], double *m, double *p, const int n) {
	//総エネルギーを計算する
	double v2, sumEnergy = 0;

	for (int i = 0; i < n; i++) {
		v2 = pow(v[i][0], 2) + pow(v[i][1], 2) + pow(v[i][2], 2);
		p[i] = 0.5 * m[i] * v2;
		sumEnergy += p[i];
	}

	return sumEnergy;
}

void runge4(double (*x)[3], double (*v)[3], double *m, double (*a)[3], double *p, const
double dt, const int n) {
	gravity(x, m, a, p, n);

	for (int i = 0; i < n; i++) {
```

```
		for (int j = 0; j < 3; j++) {
			k1[i][j] = v[i][j] * dt;
			l1[i][j] = a[i][j] * dt;

			xtmp[i][j] = x[i][j] + 0.5 * k1[i][j];
		}
	}

	gravity(xtmp, m, a, p, n);

	for (int i = 0; i < n; i++) {
		for (int j = 0; j < 3; j++) {
			k2[i][j] = (v[i][j] + 0.5 * l1[i][j]) * dt;
			l2[i][j] = a[i][j] * dt;

			xtmp[i][j] = x[i][j] + 0.5 * k2[i][j];
		}
	}

	gravity(xtmp, m, a, p, n);

	for (int i = 0; i < n; i++) {
		for (int j = 0; j < n; j++) {
			k3[i][j] = (v[i][j] + 0.5 * l2[i][j]) * dt;
			l3[i][j] = a[i][j] * dt;

			xtmp[i][j] = x[i][j] + k3[i][j];
		}
	}

	gravity(xtmp, m, a, p, n);
```

```
	for (int i = 0; i < n; i++) {
		for (int j = 0; j < n; j++) {
			k4[i][j] = (v[i][j] + l3[i][j]) * dt;
			l4[i][j] = a[i][j] * dt;

			k[i][j] = (k1[i][j] + 2.0 * k2[i][j] + 2.0 * k3[i][j] + k4[i][j]) / 6.0;
			l[i][j] = (l1[i][j] + 2.0 * l2[i][j] + 2.0 * l3[i][j] + l4[i][j]) / 6.0;

			x[i][j] += k[i][j];
			v[i][j] += l[i][j];
		}
	}
}

int main() {
	double x[NMAX][3];
	double v[NMAX][3];
	double a[NMAX][3];
	double m[NMAX];
	double p[NMAX];
	int n;
	double dt, tend;

	cerr << "input n, dt, tend, m, x, v" << endl;
	cin >> n >> dt >> tend;
	cerr << n << "¥t" << dt << "¥t" << tend << endl;
	for (int i = 0; i < n; i++) {
		cin >> m[i];
		for (int j = 0; j < 3; j++) cin >> x[i][j];
		for (int j = 0; j < 3; j++) cin >> v[i][j];
```

```
	}

	gravity(x, m, a, p, n);
	double e0 = energy(v, m, p, n);
	double tnow = 0.0;

	cout << tnow << "¥t"
		<< x[0][0] << "¥t" << x[0][1] << "¥t" << x[0][2] << "¥t"
		<< x[1][0] << "¥t" << x[1][1] << "¥t" << x[1][2] << "¥t"
		<< x[2][0] << "¥t" << x[2][1] << "¥t" << x[2][2] << "¥t";

	while (tnow < tend) {
		runge4(x, v, m, a, p, dt, n);
		double e = energy(v, m, p, n);
		tnow += dt;

		cout << tnow << "¥t"
			<< x[0][0] << "¥t" << x[0][1] << "¥t" << x[0][2] <<
"¥t"
			<< x[1][0] << "¥t" << x[1][1] << "¥t" << x[1][2] <<
"¥t"
			<< x[2][0] << "¥t" << x[2][1] << "¥t" << x[2][2] <<
"¥t"
			<< e << "¥t" << (e - e0) / e0 << endl;
	}

	return 0;
}
```

図 1 ルンゲクッタ 2 次，4 次によって 3 体問題を解くプログラム

図 2 に与えられた初期条件 2 種類を示す.

3 0.001 8.0 1.0 -0.97000436 0.2430875 0.0 0.466203685 0.43236573 0.0 1.0 0.97000436 -0.2430875 0.0 0.466203685 0.43236573 0.0 1.0 0.0 0.0 0.0 -0.93240737 -0.86473146 0.0

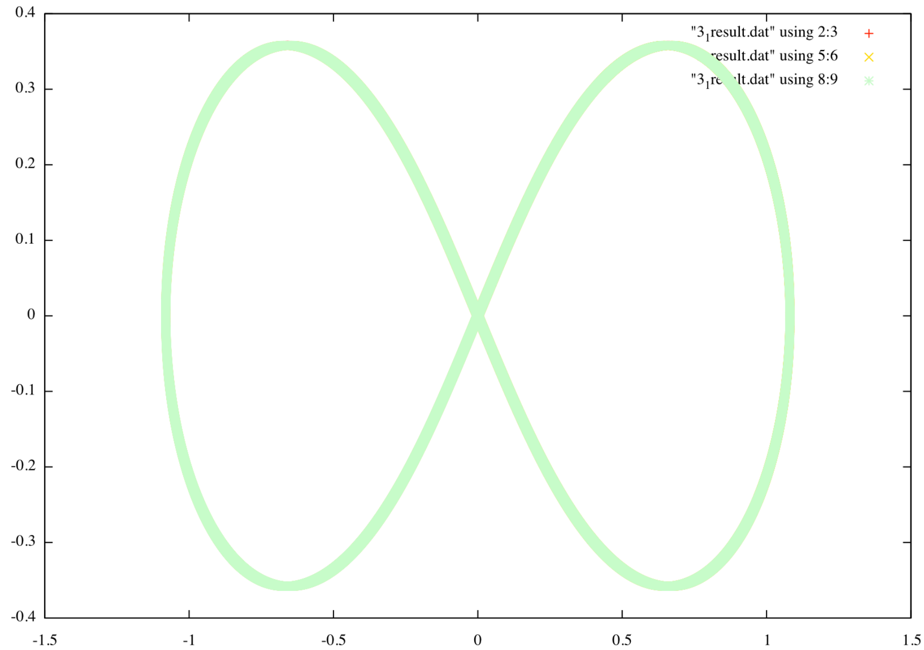
(a)初期条件 1

3 0.001 40.0 5.0 0.0 0.0 0.0 1.73205 -1.0 0.0 5.0 2.0 0.0 0.0 0.0 2.0 0.0 5.0 1.0 1.7320508 0.0 -1.73205 -1.0 0.0

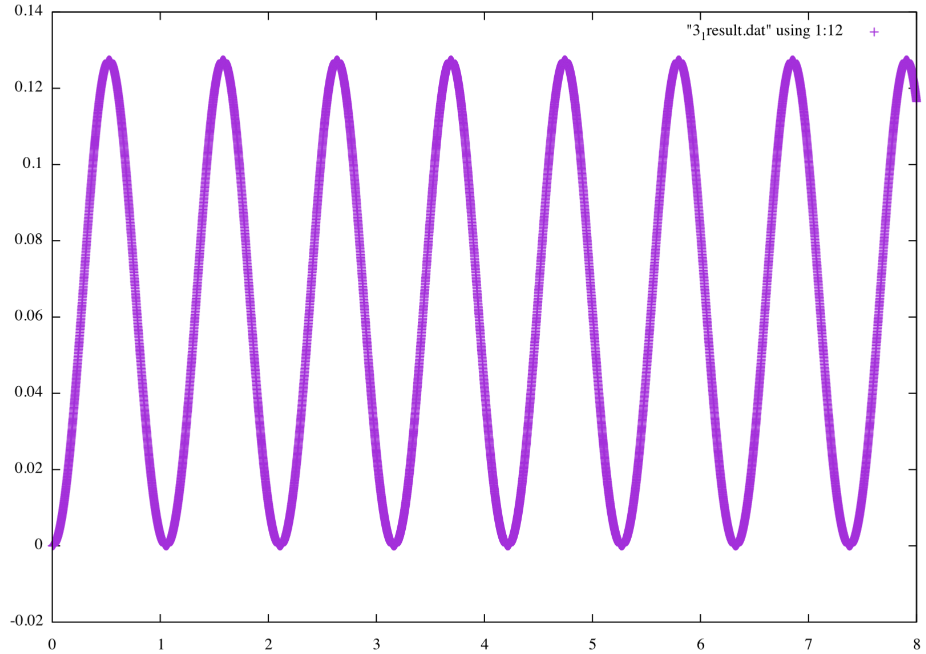
(b)初期条件 2

図 2 初期条件

図 3 にルンゲクッタ 2 次で初期条件 1 の 3 体問題を解いた結果を，図 4 にルンゲクッタ 4 次で初期条件 1 の 3 体問題を解いた結果を示す.

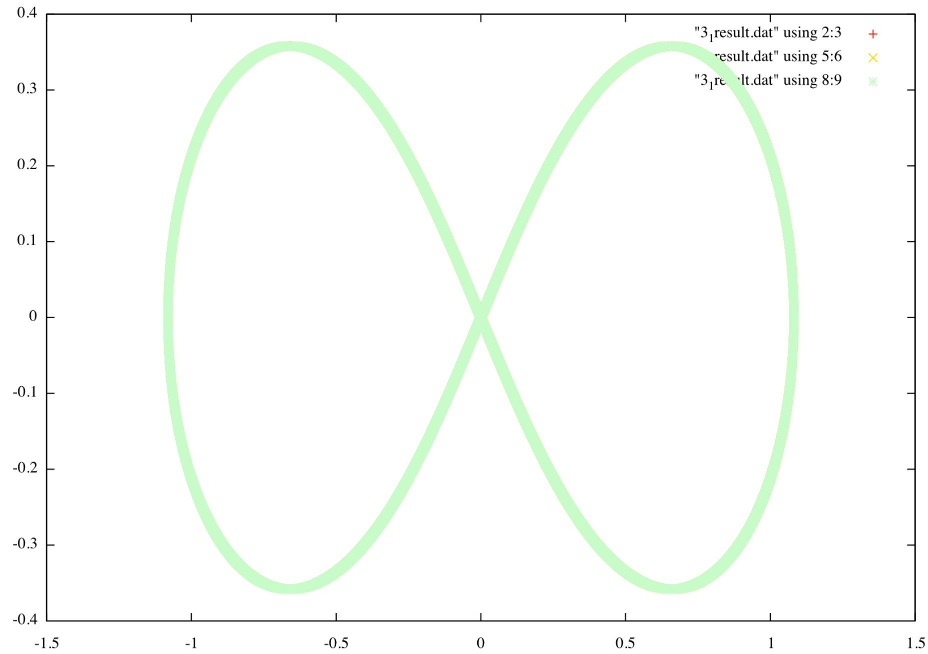


(a) $x-y$平面における 3 質点の軌道

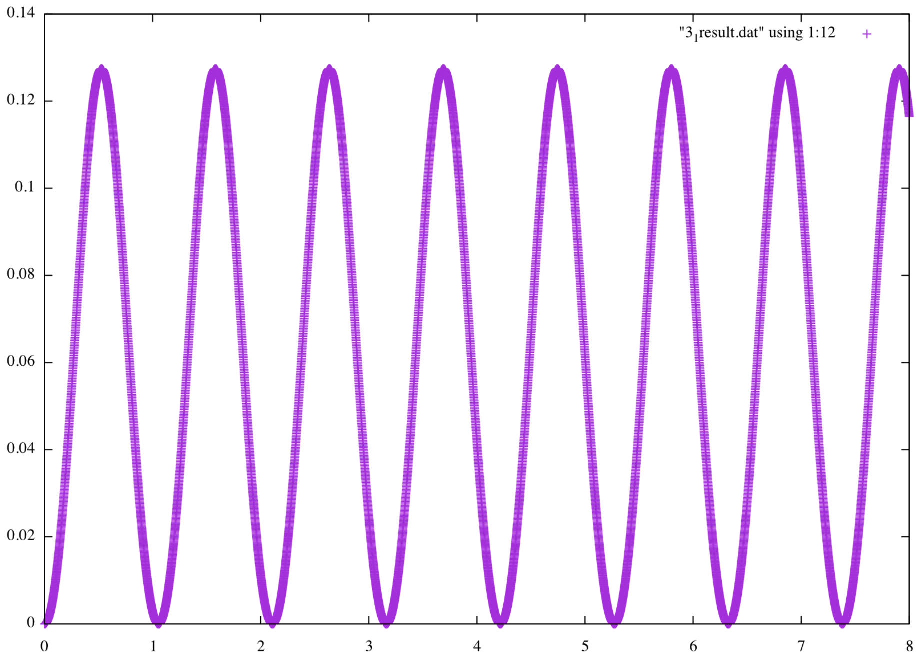


(b)エネルギー誤差の時間変化のグラフ

図 3 ルンゲクッタ 2 次による初期条件 1 の解



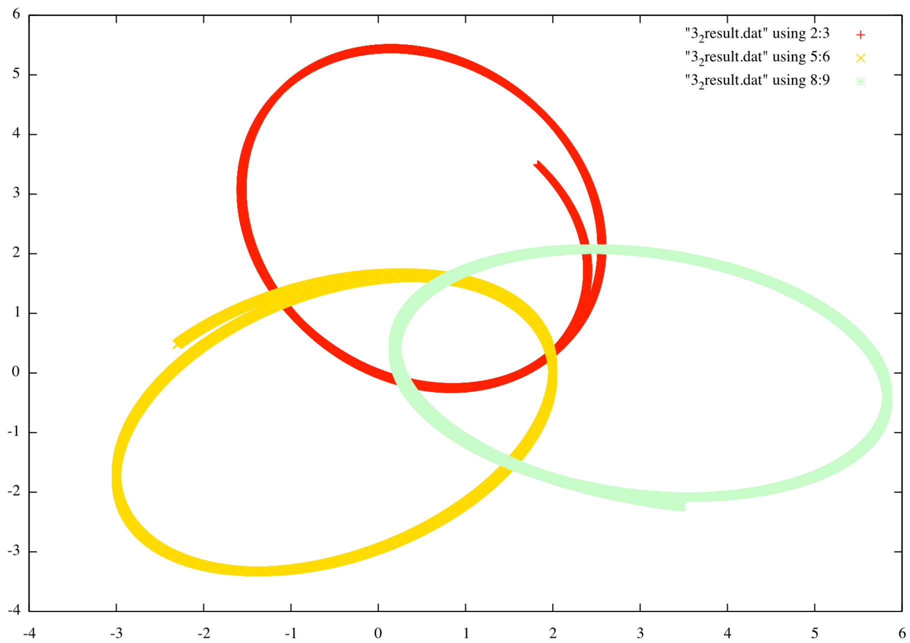
(a) $x-y$平面における 3 質点の軌道



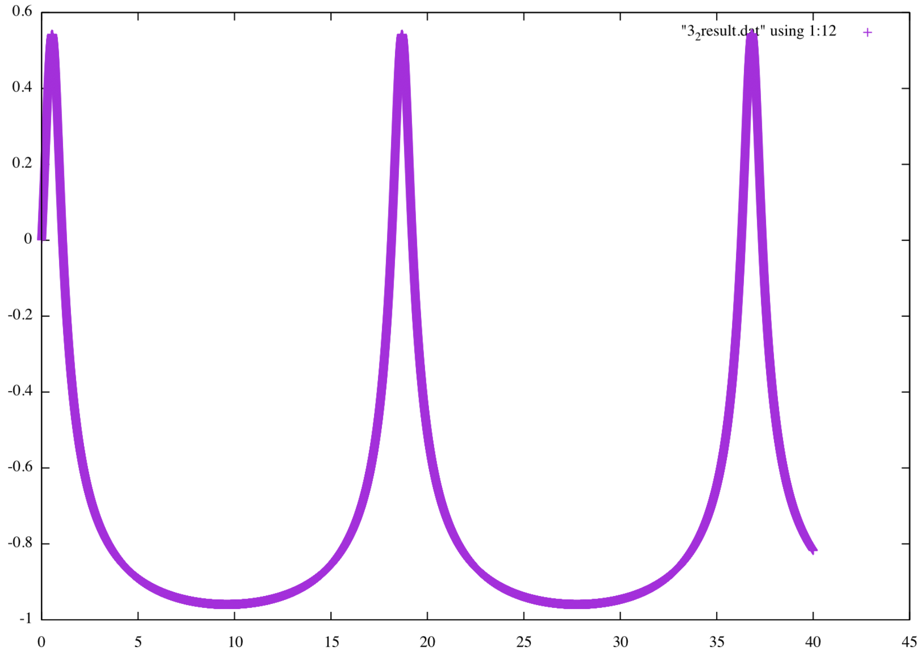
(b) エネルギー誤差の時間変化のグラフ

図 4 ルンゲクッタ 4 次による初期条件 1 の解

図 5 にルンゲクッタ 2 次で初期条件 2 の 3 体問題を解いた結果を，図 6 にルンゲクッタ 4 次で初期条件 2 の 3 体問題を解いた結果を示す．

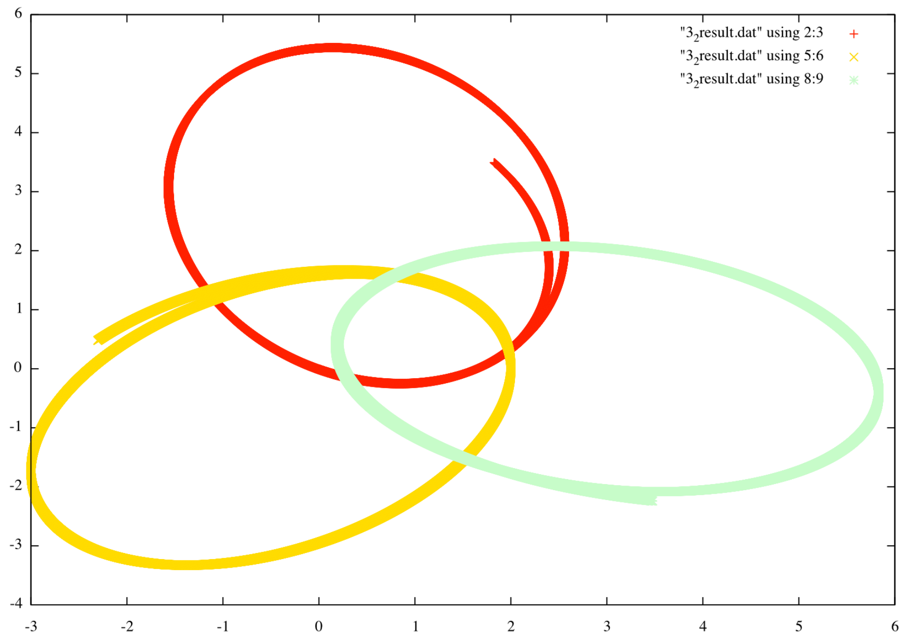


(a) $x-y$平面における 3 質点の軌道

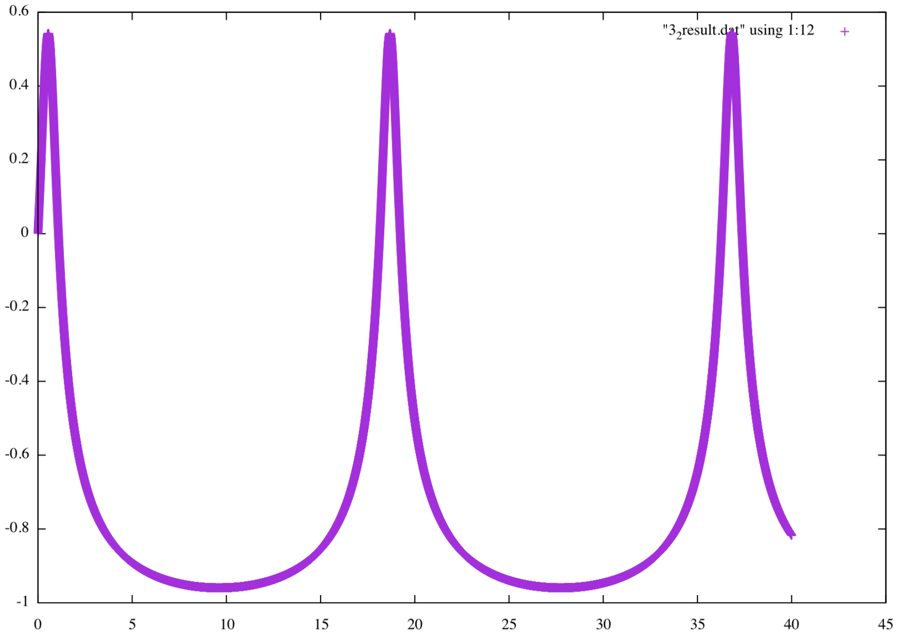


(b) エネルギー誤差の時間変化のグラフ

図 5 ルンゲクッタ 2 次による初期条件 2 の解



(a) $x - y$平面における 3 質点の軌道



(b) エネルギー誤差の時間変化のグラフ

図 6 ルンゲクッタ 4 次による初期条件 2 の解

初期条件 1 の時の結果をルンゲクッタ 2 次と 4 次とで比較してみる．エネルギー誤差の時間変化のグラフ図 3(b) と図 4(b) において，ルンゲクッタ 2 次のときは周期の最低値としてエネルギー誤差が負の値になることがあるが，4 次のときは負の値になることはない．エネルギー誤差の範囲が広いことから，エネルギー保存の計算精度は，ルンゲクッタ 2 次よりも 4 次のほうが良いといえる．

初期条件 2 の時の結果をルンゲクッタ 2 次と 4 次とで比較してみる．図 5(a) と図 6(a) のグラフにおいて，2 次の場合も 4 次の場合も軌道を表せているが，2 周目の 3 質点の軌道が外れているため，計算精度はどちらも良くないと考えられる．また，ルンゲクッタ 2 次と 4 次とで黄色の軌道がとる値の範囲が異なることがグラフから読み取れる．よって，ルンゲクッタ 2 次と 4 次両方ともで 3 質点の軌道を表すことができるが，計算の結果が多少異なることがわかる．