Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 1

Тема: «Приближенный аналитический метод Пикара в сравнении с численными методами»

Студент: Гасанзаде М.А.

Группа: ИУ7-66Б

Оценка (баллы) _______________

Преподаватель: Градов В.М.

Москва.
2020 г.

Оглавление

Введение

Во многих случаях ОДУ не интегрируются в явном виде. Общее решение в квадратурах удается получить лишь для узкого класса функций, стоящих в левых частях СОДУ и правых частях СОДУ, в связи с чем, возникает потребность в приближенных методах интегрирования ОДУ, которые, как правило, дают неточное решение ЗК или ДЗК.

Весьма условно, в зависимости от формы представления решения, эти методы подразделяются на две основные группы:

- приближенно-аналитические методы, применение которых дает приближенное решение ЗК в виде формулы;
- численные методы, когда искомая функция получается в виде таблицы, т. е. поиск решения происходит на сетке. К приближенно-аналитическим методам относятся метод последовательных приближений Пикара и метод рядов Тейлора, в которых приближенное решение ищется в виде аналитического выражения.

1. Аналитическая часть

1.1 Задача

Решить уравнение $u'(x) = u^2 + x^2$.

Используемые методы:

1. Метод Пикара 2 приближения
2. Метод Пикара 2 приближения
3. Метод ломанных
4. Метод Рунге-Кутта 2 порядка
5. Метод Рунге-Кутта 4 порядка

1.2 Метод Пикара

Первое приближение имеет вид:

$$y_1(x) := y_0 + \int_{x_0}^{x} f(\tau, y_0)\, d\tau \qquad (1)$$

Все дальнейшие приближения строятся по формуле:

$$y_{i+1}(x) := y_0 + \int_{x_0}^{x} f(\tau, y_i(\tau))\, d\tau, i = 1, 2, \ldots \qquad (2)$$

Приближения:

$u(0) = 0$

$$y^{(1)} = y(0) + \int_0^x t^2 dt = \frac{x^3}{3}$$

$$y^{(2)} = y(0) + \int_0^x [t^2 + \left(\frac{x^3}{3}\right)^2 dt = \left.\frac{t^3}{3}\right|_0^x + \left.\frac{t^7}{7*9}\right|_0^x = \frac{x^3}{3}\left[1 + \frac{x^4}{21}\right]$$

$$y^{(3)} = 0 + \int_0^x [t^2 + \left\{\frac{x^3}{3}\left(1 + \frac{x^4}{21}\right)\right\}^2 dt = \left.\frac{t^3}{3}\right|_0^x + \int_0^x \frac{x^{14}}{3969} + \frac{2x^{10}}{189} + \frac{x^6}{9} dt =$$

$$= \left.\frac{t^3}{3}\right|_0^x + \int_0^x \frac{x^{14}}{3969} dt + \int_0^x \frac{2x^{10}}{189} dt + \int_0^x \frac{x^6}{9} dt = \left.\frac{t^3}{3}\right|_0^x$$

$$= + \int_0^x \frac{x^{14}}{3969} dt + \int_0^x \frac{2x^{10}}{189} dt + \int_0^x \frac{x^6}{9} dt =$$

$$= \left.\frac{t^3}{3}\right|_0^x + \left.\frac{t^{15}}{15*3969}\right|_0^x + \left.\frac{2t^{11}}{11*189}\right|_0^x + \left.\frac{t^7}{7*9}\right|_0^x = \frac{x^3}{3} + \frac{x^{15}}{59535} + \frac{2x^{11}}{2079} + \frac{x^7}{63}$$

$$y^{(4)} = 0 + \int_0^x t^2 + \left(\frac{t^3}{3} + \frac{t^{15}}{59535} + \frac{2t^{11}}{2079} + \frac{t^7}{63}\right)^2 dt$$
$$= \int_0^x t^2 + \frac{t^{30}}{3544416225} + \frac{4t^{26}}{123773265} + \frac{662t^{22}}{45383505} + \frac{82t^{18}}{1964655} + \frac{13t^{14}}{14553} + \frac{2t^{10}}{189} + \frac{t^6}{9} dt =$$
$$= \frac{x^{31}}{109876902975} + \frac{4 * x^{27}}{3341878155} + \frac{662x^{23}}{10438212015} + \frac{82x^{19}}{37328445} + \frac{13x^{15}}{218295} + \frac{2x^{11}}{2079} + \frac{x^7}{63} + \frac{x^3}{3}$$

Теорема Пикара доставляет не только условия, при которых процесс (1), (2) сходится, но и оценку погрешности n-го приближения.

В силу трудностей с нахождением первообразных в чистом виде, метод (1)–(2) редко реализуем. Чаще применяется его псевдоаналитический вариант, когда в формулах (1) и (2) интегралы заменяются на квадратурные суммы. Но главную известность метод получил как инструмент для доказательства теоремы Пикара.

1.3 Метод ломанных (явная схема)

Это простейший численный метод. В практике вычислений он употребляется очень редко из-за невысокой точности. Но на его примере удобно пояснить способы построения и исследования численных методов.

Рассмотрим задачу Коши и выберем на отрезке $[\xi, X]$ некоторую сетку $\{x_n, 0 \le n \le N\}$ значений аргумента так, чтобы выполнялись соотношения $\xi = x_0 < x_1 < x_2 < \dots < x_N = X$ (сетка может быть неравномерной). Разлагая решение и u(x) по формуле Тейлора на интервале сетки $x_n \le x \le x_{n+1}$ и обозначая и u(xn) = un получим

$$u_{n+1} = u_n + h_n u_n' + \frac{1}{2} h_n^2 u_n'' + \dots, \quad h_n = x_{n+1} - x_n. \qquad (3)$$

Стоящие в правой части производные можно найти, дифференцируя уравнение $u'(x) = f\big(x, u(x)\big),\ f\xi \le x \le X,\ fu(\xi) = \eta$ требуемое число раз: (4)

$$u' = f(x, u), \quad u'' = \frac{d}{dx} f(x, u) = f_x + f f_u \qquad (5)$$

и т. д. В принципе, если (4) имеет непрерывные производные по совокупности аргументов, то в разложении (3) можно удержать члены вплоть до $O(h^{q+1})$

Однако использовать для расчетов формулу 3 с большим числом членов невыгодно. Во-первых, даже при сравнительно простой правой части выражения для производных могут оказаться громоздкими. Во-вторых, если правая часть известна лишь приближенно, то находить ее производные нежелательно. В простейшем случае, подставляя (5) в (3) и ограничиваясь только первым членом разложения, получим схему ломаных

$$y_{n+1} = y_n + h_n f(x_n,\ y_n), \quad h_n = x_{n+1} - x_n. \qquad (6)$$

Поскольку при такой замене можно найти только приближенные значения искомой функции в узлах, то будем обозначать эти значения через y_n в отличие от точных значений $u_n = u(x_n)$ Для численного расчета по схеме ломаных достаточно задать начальное значение $y_0 = \eta$. Затем по формуле (6) последовательно вычисляем величины $y_1, y_2, y_3, \dots, y_N$

1.4 Неявная схема

$$y_{n+1} = y_n + h * [f(x_{n+1}; y_{n+1})] \qquad (7)$$

Решая это алгебраическое уравнение, можно определить y_{n+1} которое и будет приближенным значением искомого решения и $u(x_n)$. Схема (7) имеет второй порядок точности, допускает счет неравномерным шагом, не требует специальных приемов для начала счета. Но у этой схемы есть серьезные недостатки. Во-первых, неизвестно, имеет ли уравнение (7) вещественный корень, т. е. разрешима ли задача. Можно привести пример, когда при большом шаге корня нет. Пусть $f(x,u) = u^2$ и $u(0) = 1$; тогда на первом шаге
$y_1 = 1 + \frac{1}{2}h(1 + y_1^2)$ и при $h > (1+\sqrt{2})$ вещественного корня нет.

Во-вторых, даже если корень есть, то как его найти? Метод Ньютона применять нежелательно, так как для этого надо дифференцировать $f(x,u)$. Метод деления пополам не обобщается на системы уравнений. Остается метод последовательных приближений.

1.5 Метод Рунге-Кутта

Можно строить схемы различного порядка точности. Например, схема ломаных (5) есть схема Рунге—Кутта первого порядка точности. Наиболее употребительны схемы четвертого порядка точности, образующие семейство четырехчленных схем. Приведем без вывода ту из них, которая записана в большинстве стандартных программ ЭВМ:

$$y_{n+1} = y_n + (k_1 + 4k_2 + k_3)/6\ ,$$
$$k_1 = h_n\phi(x_n, y_n)\ , \quad k_2 = h_n\phi(x_n + h_n/2, y_n + k_1/2)\ ,$$
$$k_3 = h_n\phi(x_n + h_n, y_n - k_1 + 2k_2)\ ,$$

Формулы более высокого порядка точности практически не употребляются. Пятичленные формулы имеют всего лишь четвертый порядок точности; шестичленные имеют шестой порядок, но слишком громоздки. Кроме того, высокий порядок реализуется лишь при наличии у правой части непрерывных производных соответствующего порядка.

Схемы Рунге—Кутта имеют ряд важных достоинств.

1) Все они (кроме схемы ломаных) имеют хорошую точность.
2) Они являются явными, т. е. значение y_{n+1} вычисляется по ранее найденным значениям за определенное число действий по определенным формулам.
3) Все схемы допускают расчет переменным шагом; значит, нетрудно уменьшить шаг там, где функция быстро меняется, и увеличить его в обратном случае.
4) 4) Для начала расчета достаточно выбрать сетку x_n и задать значение $y_0 = \eta$; далее вычисления идут по одним и тем же формулам. Все эти свойства схем очень ценны при расчетах на ЭВМ.

2. Листинг

```
from prettytable import PrettyTable
#addimlarla ireli
h = 0.00001

#aslinda x**2+u**2 cauchy problemidir
def function(x,u): return pow(x,2)+pow(u,2)

def Picar3(x):
    y = pow(x, 3) / 3
    tmp = 1 + pow(x, 4) / 21
    tmp += 2 * pow(x, 3) / (693)
    tmp += pow(x, 12) / (19845)
    return y*tmp

def Picar4(x):
    y = pow(x, 31)/109876902975
    y += 2 * pow(x, 23) / 86266215
    y += 2 * pow(x, 22) / 1361505915
    y += 2 * pow(x, 19) / 3393495
    y += pow(x, 15) / 59535
    y += 2 * pow(x, 14) / 916839
    y += pow(x, 13) / 56189133
    y += 2 * pow(x, 11)/2079
    y += 2 * pow(x, 10)/31185
    y += 2 * pow(x, 7)/63
    y += pow(x, 3) / 3
    return y

#sayisal yontem (coklu cizgi yontemi)
def polyline(x, y): return (y + h * function(x, y))

def Runge2(x, y):
    return y + h * function(x + h / 2, y + h / 2 * function(x, y))

def Runge4(x, y):
    K1 = function(x, y)
    K2 = function(x + h / 2, y + h * K1 / 2)
    K3 = function(x + h / 2, y + h * K2 / 2)
    K4 = function(x + h, y + h * K3)
    return y + h / 6 * (K1 + 2 * K2 + 2 * K3 + K4)

def main():
    #sinirlar
    x = 0.0
    maxX = 2.0
    poly = 0.0
    run2 = 0.0
    run4 = 0.0
    tb = PrettyTable([" X ","Picard 3", "Picard 4", "polyline явный",
"R2nd", "R4th"])

    while(x <= maxX):
        tb.add_row([round(x, 7), round(Picar3(x), 7), round(Picar4(x),
7), round(poly, 7), round(run2, 7), round(run4, 7)])
        poly = polyline(x, poly)
        run2 = Runge2(x, run2)
        run4 = Runge4(x, run4)
        x += h
    print(tb, "\n")
    return 0
main()
```

3. Результат работы программы

Пример работы программы будет представлен ниже на *рис. 1 и 2*, при значении шага = 0.01

============== RESTART: C:\Users\pbhac\Desktop\Mat_mod\math1.py ==============

X	Picard 3	Picard 4	polyline явный	R2nd	R4th
0.0	0.0	0.0	0.0	0.0	0.0
0.01	3e-07	3e-07	0.0	3e-07	3e-07
0.02	2.7e-06	2.7e-06	1e-06	2.5e-06	2.7e-06
0.03	9e-06	9e-06	5e-06	8.8e-06	9e-06
0.04	2.13e-05	2.13e-05	1.4e-05	2.1e-05	2.13e-05
0.05	4.17e-05	4.17e-05	3e-05	4.13e-05	4.17e-05
0.06	7.2e-05	7.2e-05	5.5e-05	7.15e-05	7.2e-05
0.07	0.0001143	0.0001143	9.1e-05	0.0001138	0.0001143
0.08	0.0001707	0.0001707	0.00014	0.00017	0.0001707
0.09	0.000243	0.000243	0.000204	0.0002423	0.000243
0.1	0.0003333	0.0003333	0.000285	0.0003325	0.0003333
0.11	0.0004437	0.0004437	0.000385	0.0004428	0.0004437
0.12	0.000576	0.000576	0.000506	0.000575	0.000576
0.13	0.0007323	0.0007324	0.00065	0.0007313	0.0007323
0.14	0.0009147	0.0009147	0.000819	0.0009135	0.0009147
0.15	0.001125	0.0011251	0.001015	0.0011238	0.001125
0.16	0.0013654	0.0013654	0.00124	0.001364	0.0013654
0.17	0.0016378	0.0016378	0.001496	0.0016363	0.0016377
0.18	0.0019441	0.0019442	0.0017851	0.0019426	0.0019441
0.19	0.0022865	0.0022866	0.0021091	0.0022849	0.0022865
0.2	0.0026669	0.0026671	0.0024701	0.0026652	0.0026669
0.21	0.0030874	0.0030876	0.0028702	0.0030855	0.0030873
0.22	0.0035498	0.0035501	0.0033113	0.0035479	0.0035497
0.23	0.0040563	0.0040567	0.0037954	0.0040543	0.0040562
0.24	0.0046089	0.0046095	0.0043245	0.0046067	0.0046087
0.25	0.0052095	0.0052103	0.0049007	0.0052072	0.0052093
0.26	0.0058602	0.0058612	0.005526	0.0058578	0.0058599
0.27	0.006563	0.0065643	0.0062023	0.0065604	0.0065627
0.28	0.0073199	0.0073216	0.0069317	0.0073171	0.0073195
0.29	0.008133	0.0081351	0.0077161	0.00813	0.0081324
0.3	0.0090042	0.0090069	0.0085577	0.009001	0.0090035
0.31	0.0099356	0.0099391	0.0094585	0.0099321	0.0099347
0.32	0.0109292	0.0109336	0.0104204	0.0109254	0.0109281
0.33	0.011987	0.0119925	0.0114454	0.011983	0.0119858
0.34	0.0131112	0.013118	0.0125358	0.0131068	0.0131097
0.35	0.0143036	0.0143121	0.0136933	0.0142989	0.0143019
0.36	0.0155665	0.0155769	0.0149202	0.0155614	0.0155645
0.37	0.0169019	0.0169145	0.0162184	0.0168963	0.0168994
0.38	0.0183117	0.018327	0.0175901	0.0183056	0.0183089
0.39	0.0197982	0.0198166	0.0190372	0.0197915	0.0197948
0.4	0.0213633	0.0213854	0.0205618	0.021356	0.0213594
0.41	0.0230091	0.0230356	0.022166	0.0230011	0.0230046
0.42	0.0247379	0.0247693	0.0238519	0.0247291	0.0247327
0.43	0.0265516	0.0265887	0.0256216	0.0265419	0.0265456
0.44	0.0284523	0.0284962	0.0274772	0.0284417	0.0284455
0.45	0.0304423	0.0304938	0.0294207	0.0304306	0.0304345
0.46	0.0325236	0.0325839	0.0314544	0.0325107	0.0325147
0.47	0.0346985	0.0347688	0.0335803	0.0346842	0.0346883
0.48	0.036969	0.0370507	0.0358005	0.0369533	0.0369575
0.49	0.0393373	0.0394321	0.0381174	0.0393201	0.0393244
0.5	0.0418057	0.0419152	0.0405329	0.0417868	0.0417911
0.51	0.0443764	0.0445026	0.0430493	0.0443555	0.04436
0.52	0.0470515	0.0471965	0.0456689	0.0470286	0.0470332
0.53	0.0498335	0.0499996	0.0483937	0.0498083	0.049813
0.54	0.0527244	0.0529143	0.0512261	0.0526968	0.0527016
0.55	0.0557266	0.0559432	0.0541684	0.0556964	0.0557013
0.56	0.0588425	0.0590888	0.0572227	0.0588094	0.0588145
0.57	0.0620743	0.0623538	0.0603915	0.0620381	0.0620433
0.58	0.0654244	0.065741	0.0636769	0.0653849	0.0653902
0.59	0.0688953	0.069253	0.0670815	0.0688522	0.0688576
0.6	0.0724892	0.0728926	0.0706075	0.0724423	0.0724479

Рисунок 1. Пример работы программы - Начало

0.61	0.0762088	0.0766627	0.0742573	0.0761577	0.0761634
0.62	0.0800563	0.0805662	0.0780335	0.0800009	0.0800067
0.63	0.0840344	0.0846061	0.0819384	0.0839743	0.0839803
0.64	0.0881456	0.0887854	0.0859745	0.0880805	0.0880866
0.65	0.0923924	0.0931072	0.0901444	0.092322	0.0923283
0.66	0.0967774	0.0975748	0.0944507	0.0967015	0.096708
0.67	0.1013034	0.1021913	0.0988959	0.1012216	0.1012282
0.68	0.105973	0.1069602	0.1034827	0.1058851	0.1058918
0.69	0.1107888	0.1118848	0.1082138	0.1106945	0.1107014
0.7	0.1157538	0.1169687	0.1130919	0.1156527	0.1156599
0.71	0.1208707	0.1222154	0.1181198	0.1207626	0.1207699
0.72	0.1261423	0.1276288	0.1233003	0.126027	0.1260345
0.73	0.1315716	0.1332126	0.1286363	0.1314489	0.1314566
0.74	0.1371616	0.1389706	0.1341308	0.1370313	0.1370392
0.75	0.1429152	0.1449071	0.1397867	0.1427771	0.1427852
0.76	0.1488356	0.1510261	0.1456071	0.1486897	0.148698
0.77	0.1549259	0.1573318	0.1515951	0.154772	0.1547805
0.78	0.1611892	0.1638287	0.1577539	0.1610274	0.1610362
0.79	0.1676289	0.1705214	0.1640868	0.1674593	0.1674683
0.8	0.1742483	0.1774145	0.1705971	0.174071	0.1740803
0.81	0.1810506	0.1845128	0.1772881	0.180866	0.1808756
0.82	0.1880396	0.1918214	0.1841634	0.187848	0.1878578
0.83	0.1952185	0.1993453	0.1912266	0.1950205	0.1950306
0.84	0.2025912	0.2070899	0.1984812	0.2023874	0.2023978
0.85	0.2101611	0.2150607	0.2059312	0.2099525	0.2099632
0.86	0.2179323	0.2232633	0.2135803	0.2177198	0.2177308
0.87	0.2259084	0.2317036	0.2214324	0.2256934	0.2257047
0.88	0.2340935	0.2403877	0.2294917	0.2338774	0.2338891
0.89	0.2424915	0.2493218	0.2377624	0.2422762	0.2422883
0.9	0.2511067	0.2585123	0.2462487	0.2508942	0.2509067
0.91	0.2599433	0.2679661	0.2549551	0.259736	0.2597489
0.92	0.2690055	0.27769	0.2638861	0.2688064	0.2688197
0.93	0.2782979	0.2876913	0.2730465	0.2781101	0.2781238
0.94	0.2878249	0.2979773	0.282441	0.2876522	0.2876664
0.95	0.2975913	0.3085558	0.2920748	0.2974379	0.2974526
0.96	0.3076019	0.3194347	0.3019528	0.3074726	0.3074878
0.97	0.3178614	0.3306224	0.3120806	0.3177618	0.3177775
0.98	0.328375	0.3421274	0.3224635	0.3283111	0.3283274
0.99	0.3391478	0.3539586	0.3331074	0.3391266	0.3391435

Рисунок 2. Пример работы программы - Конец

Заключение

В общем к приближенно-аналитическим методам относятся метод последовательных приближений Пикара и метод рядов Тейлора, в которых приближенное решение ищется в виде аналитического выражения. Их преимущество перед численными методами в том, что они выдают “непрерывное” решение, в то время как численными получают дискретные решения (решения на сетке), а недостатком приближенно-аналитических методов, в сравнении с численными, являются более высокие требования к правым частям СОДУ (типа существования производных высоких порядков у правых частей), в то время как численные методы можно применять к СОДУ даже всего лишь с непрерывной правой частью.

А на практике же метод Пикара используется довольно редко, так как интегралы, которые необходимо вычислять при построении очередных приближений, чаще всего аналитически не находятся, а применение для их вычисления численных методов так усложняет решение, что становится гораздо удобнее непосредственно применять численные методы.

Список использованной литературы

1. Методы приближенного решения задачи коши URL: http://www.apmath.spbu.ru/ru/staff/grigorieva/glava1-2.pdf

2. Градов В.М. Курс лекций по Моделированию - 2020