http://mmg.tversu.ru ISSN 2311-1275

Mathematical Modelling and Geometry

Volume 14, No 2, p. 1 – 10 (2026) doi:10.26456/mmg/2026-1421

On finding the integrating factor of a second-order ordinary differential equation in symbolic form

Ahmad Maarouf[1,a] **and Ali Baddour**[1]

[1] RUDN University, 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

e-mail: [a] ahmad-kh-maarouf@hotmail.com

Received 28 March 2026, in final form 30 April. Published 12 May 2026.

Abstract. The problem of integrating a second-order ordinary differential equation $f(x,y,y')y''+g(x,y,y')=0$ is considered in symbolic form. First, it is shown how to determine in a finite number of steps whether there exists a function $u(x,y,y')$ such that $f(x,y,y')y''+g(x,y,y')=du$, and how to find such a function. Following Cheb-Terrab's proposal and by analogy with the case of first-order equations, such equations are called exact. Then, the issue of finding an integrating factor in a finite number of steps is considered, i.e., finding a function μ such that the equation $\mu\cdot(f(x,y,y')y''+g(x,y,y'))=0$ is exact. As in the case of first-order equations, the factor can be calculated explicitly under the assumption that it depends on one of the three variables x, y or y'. The case $\mu(y')$ is considered in detail.

Keywords: ordinary differential equations, symbolic integration, integrating multiplier

PACS numbers: 07.05.Tp, 02.60.Pn, 02.70.Bf

---

1. Introduction

Methods of symbolic integration of first-order ordinary differential equations are based, directly or indirectly, on the calculation of the integrating factor [1]. Immediately after creating the first symbolic integrator, J. Moses began developing software for integrating differential equations, and he based it on algorithms for calculating factors that depend on only one variable [2, 3]. Calculating the factor and the integral requires several symbolic calculations of the indefinite integrals.

This concept was extended to second-order differential equations of the form

$$y'' = g(x, y, y') \tag{1}$$

in the 1990s, as part of a project to develop symbolic integration methods and implement them in the Maple computer algebra system. An algorithm was developed that can determine whether Eq. (1) admits a factor of the form $\mu(x, y)$, $\mu(x, y')$, or $\mu(y, y')$ and, if so, calculate the intermediate integral [4].

A recent article by AlAhmad addressed the problem of finding the integrating factor for a second-order differential equation of the form

$$a_2(x, y, y')y'' + a_1(x, y, y')y' + a_0(x, y, y') = 0. \tag{2}$$

It is easy to see that, given a second-order equation, the coefficients a_1 and a_2 cannot be uniquely determined. Success or failure in calculating the factor turns out to be related to the choice of representation of the differential equation in question in the form (2). Therefore, the conditions for the existence of an integrating factor found in the article are sufficient, but not necessary.

However, this in no way diminishes the importance of the question posed in this article. Let a second-order equation be given, linear with respect to the second derivative:

$$f(x, y, y')y'' + g(x, y, y') = 0. \tag{3}$$

Following [4, 5], we assume that Eq. (3) is called exact if there exists a function $u(x, y, y')$ such that

$$f(x, y, y')y'' + g(x, y, y') = \frac{du}{dx}$$

for any smooth function $y(x)$. Then, for any solution $y(x)$ of Eq.(3)

$$u(x, y, y') = \text{const}.$$

We will call u the first integral of the original differential equation (3). We will call the function $\mu(x, y, y')$ an integrating factor if the equation

$$\mu \cdot (f(x, y, y')y'' + g(x, y, y')) = 0$$

is exact. Moreover, there exists a function u such that

$$f(x, y, y')y'' + g(x, y, y') = \frac{1}{\mu}\frac{du}{dx}.$$

For the special case of (1), when $f = 1$, Ref. [4] provides algorithms for finding integrating factors of the form $\mu(x, y)$, $\mu(x, y')$, or $\mu(y, y')$. Equation (3) can be reduced to the form (1) by dividing by $f(x, y, y')$. However, the absence of, say, a factor of the form $\mu(y, y')$ in this transformed equation in no way implies the absence of such a factor in the original equation. The goal of this paper is to obtain conditions for the exactness of a second-order equation and the existence of a factor depending on one argument, and then implement these conditions in the Sage computer algebra system.

2. Integration of an Exact 1-Form

As an auxiliary problem, we consider the integration of an exact 1-form. As is known, [6], the 1-form

$$f_1(x_1, \dots, x_n)dx_1 + \dots + f_n(x_1, \dots, x_n)dx_n$$

is exact if and only if the conditions

$$\frac{\partial f_i}{\partial x_j} - \frac{\partial f_j}{\partial x_i} = 0, \quad i \neq j \tag{4}$$

are fulfilled. In this case, there exists a function u of variables $x_1, \dots, x_n$, called the potential, such that

$$f_1(x_1, \dots, x_n)dx_1 + \dots + f_n(x_1, \dots, x_n)dx_n = du.$$

In Sage, it is easy to write a function that checks whether the conditions (4) are satisfied and, if so, calculates u, for example, using the formula

$$u = \int_0^1 (f_1(x_1 t, \dots, x_n t)x_1 + \dots + f_n(x_1, \dots, x_n)x_n)\, dt. \tag{5}$$

Let us present its text for completeness:

```
def pot(fs,xs):
    for [(f1,x1),(f2,x2)] in Combinations(zip(fs,xs),2):
        if diff(f1,x2)-diff(f2,x1)!=0:
            return False
    else:
        var("t")
        S=[xx==t*xx for xx in xs]
        g=sum([ff.subs(S)*xx for (ff,xx) in zip(fs,xs)])
        return integral(g,(t,0,1))
```

This function works reliably for those cases where f_i are polynomials. In the case where f_i are elementary functions, the test of (4) may be performed incorrectly, since there is no algorithm for checking the equality of an elementary function to zero. Difficulties may arise with the calculation of the integral (5) due to the fact that the Risch algorithm is very incompletely implemented [7]. Often, instead of a simple rational potential, a very complex expression containing the Γ-function is obtained.

3. Exact second-order differential equation

Given a differential equation (3), we need to determine whether there exists a function $u(x,y,p)$ such that

$$\frac{du(x,y,y')}{dx} = f(x,y,y')y'' + g(x,y,y').$$

For convenience, in cases where we need to consider the expression as a function of three variables x, y, y', we will replace y' with p. Comparing $fy'' + g$ and

$$\frac{du}{dx} = u_p y'' + u_y y' + u_x,$$

we obtain two linear differential equations

$$\frac{\partial u}{\partial p} = f(x,y,p), \quad \frac{\partial u}{\partial y}p + \frac{\partial u}{\partial x} = g(x,y,p)$$

or

$$X_1(u) = f, \quad X_2(u) = g.$$

Thus, for u we obtain a system of two quasilinear first-order partial differential equations [6]. These equations are not in involution and therefore have a nontrivial differential consequence

$$[X_1, X_2](u) = X_2(f) - X_1(g).$$

Thus, for u, we obtain a system of 3 equations:

$$\begin{cases} \dfrac{\partial u}{\partial p} = f, \\ \dfrac{\partial u}{\partial y}p + \dfrac{\partial u}{\partial x} = g, \\ \dfrac{\partial u}{\partial y} = \dfrac{\partial g}{\partial p} - p\dfrac{\partial f}{\partial y} - \dfrac{\partial f}{\partial x} \end{cases} \tag{6}$$

The system (6) can be considered a SLAE with respect to the derivatives of u. This gives explicit expressions for derivatives:

$$\begin{cases} \dfrac{\partial u}{\partial x} = p^2 \dfrac{\partial f}{\partial y} + p \dfrac{\partial f}{\partial x} - \dfrac{\partial g}{\partial p} + g, \\ \dfrac{\partial u}{\partial y} = \dfrac{\partial g}{\partial p} - p \dfrac{\partial f}{\partial y} - \dfrac{\partial f}{\partial x}, \\ \dfrac{\partial u}{\partial p} = f. \end{cases} \tag{7}$$

Thus, we obtain the following theorem.

Theorem 1. *The differential equation* (3) *is exact if and only if the 1-form*

$$\left(p^2 \frac{\partial f}{\partial y} + p \frac{\partial f}{\partial x} - \frac{\partial g}{\partial p} + g\right) dx + \left(\frac{\partial g}{\partial p} - p \frac{\partial f}{\partial y} - \frac{\partial f}{\partial x}\right) dy + f dp \tag{8}$$

is exact. In this case, the first integral of Eq. (3) *is the potential of this form.*

The proven theorem allows us to construct a function that returns either the first integral, when the given equation is exact, or False.

```
def exact(ode):
    var("y,p,q")
    Sy=[Y==y, diff(Y,x)==p, diff(Y,x,2)==q]
    eq=ode.subs(Sy)
    f=diff(eq,q)
    g=eq.subs(q=0)
    ux = p*(diff(f, y) + diff(f, x) - diff(g, p)) + g
    uy = -p*diff(f, y) - diff(f, x) + diff(g, p)
    up = f
    fs=[ux,uy,up]
    xs=[x,y,p]
    if test(fs,xs):
        return pot(fs,xs).subs(SY)
    else:
        return False
```

Let us give a few examples.

Example 1. *If we obtain a second-order equation by differentiating a first-order equation of the form* $u(x, y, y') = C$*, then the function* `exact` *returns the first integral of a second-order equation, i.e.,* u*:*

```
ode=diff(diff(Y,x) + (3*Y^4 - 1)*x,x)
exact(ode)
3*x*Y(x)^4 - x + diff(Y(x), x)
```

Example 2. *An arbitrary second-order differential equation is not exact:*

```
ode=diff(Y,x,2)+x*Y^2
exact(ode)
False
```

4. A factor of the form $\mu(y')$

Given a differential equation (3), we need to find out whether there exist functions $\mu(y')$ and $u(x, y, p)$ such that

$$\frac{1}{\mu(y')}\frac{du(x,y,y')}{dx} = f(x,y,y')y'' + g(x,y,y').$$

Moses [3] and Singer [8] have shown that the factor is usually an exponential function, so we will immediately look for it in the form

$$\mu = e^{h(y')}. \tag{9}$$

Let us write out the conditions for the existence of a factor of this type, considering f and g to be arbitrary functions of x, y, y'. We need to verify that the equation

$$\mu f(x,y,y')y'' + \mu g(x,y,y') = 0$$

is exact, that is, that the form (8), after replacing f, g with $\mu f, \mu g$, becomes exact. Let us compose a function for this:

```
def factor():
    var("y,p,q")
    H=function('H')(p)
    mu=exp(H)
    SY=[y==Y, p==diff(Y,x)]
    Sy=[Y==y, diff(Y,x)==p, diff(Y,x,2)==q]
    f=function('f')(x,y,p)*mu
    g=function('g')(x,y,p)*mu
    ux = p*(diff(f, y) + diff(f, x) - diff(g, p)) + g
    uy = -p*diff(f, y) - diff(f, x) + diff(g, p)
    up = f
    fs=[ux,uy,up]
    xs=[x,y,p]
```

```
eqs=[]
for [(f1,x1),(f2,x2)] in Combinations(zip(fs,xs),2):
    eqs.append(diff(f1,x2)-diff(f2,x1))
var("h,hp,hpp")
Su=[H==h, diff(H,p)==hp, diff(H,p,p)==hpp]
return [(eq.subs(Su)/e^h).full_simplify() for eq in eqs]
```

This will return us a system of three equations for h and its derivatives. Fortunately, the first of them contains only h':

$$2p\frac{\partial^2 f}{\partial x\partial y}+p\frac{\partial^2 f}{\partial y^2}-h'\frac{\partial g}{\partial x}-(h'p-1)\frac{\partial g}{\partial y}-p\frac{\partial g}{\partial p}+\frac{\partial^2 f}{\partial x^2}-\frac{\partial^2 g}{\partial x\partial y}=0. \tag{10}$$

Generally speaking, h' can be found from this relation. If the resulting expression depends only on p, then the factor is equal to

$$\mu=e^{\int h'(p)dp}.$$

It should be noted that this integral is defined up to an additive constant, so we define the factor up to a multiplicative constant. Of course, multiplying a differential equation by a constant does not change its accuracy, so we will omit the multiplicative constant from the factor. By calculating $\mu(p)$ explicitly, we can check whether this function is a factor or not.

In Eq. (10), the desired value h' enters with the coefficient

$$\frac{\partial g}{\partial x}+p\frac{\partial g}{\partial y}.$$

Therefore, we must also consider the special case when

$$\frac{\partial g}{\partial x}+p\frac{\partial g}{\partial y}=0 \tag{11}$$

and, therefore,

$$g=G(px-y,p).$$

In this case, Eq. (10) itself turns into a condition on f and g, and from the other two equations of the same system as this equation, we can obtain the corollary

$$-h'p^2\frac{\partial f}{\partial y}+h'p\frac{\partial f}{\partial y}-p^2\frac{\partial^2 f}{\partial y\partial p}-2p\frac{\partial f}{\partial y}+p\frac{\partial^2 f}{\partial y\partial p}+\frac{\partial f}{\partial y}=0,$$

from which we can again find h' and calculate the multiplier using the method described above.

It remains to consider the very special case when the coefficient of h' in this equation also vanishes, meaning that two conditions are simultaneously satisfied:

$$\frac{\partial g}{\partial x}+p\frac{\partial g}{\partial y}=0,\quad \frac{\partial f}{\partial y}=0. \tag{12}$$

We have never encountered this case, but in the code below we have created a special branch for it.

In our function `factorp`, we have explicitly written out the solutions of the equations for h'.

```
def factorp(ode):
    var("y,p,q")
    SY=[y==Y, p==diff(Y,x)]
    Sy=[Y==y, diff(Y,x)==p, diff(Y,x,2)==q]
    eq=ode.subs(Sy)
    g=eq.subs(q=0)
    f=diff(eq,q)
    if diff(g,x)+p*diff(g,y)!=0:
        D=[hp == (2*p*diff(f, x, y) + p*diff(f, y, y) \
        - p*diff(g, y, p) + diff(f, x, x) - diff(g, x, p) \
        + diff(g, y))/(p*diff(g, y) + diff(g, x))]
    else:
        if diff(f, y)!=0:
            D=[hp== -((2*p - 1)*diff(f, y) \
            + (p^2 - p)*diff(f, y, p))/((p^2 - p)*diff(f, y))]
        else:
            print('Warning! Very special case.' )
            return False
    try:
        mu=exp(integral(hp.subs(D),p)).subs(SY)
        u=exact(ode*mu)
    except:
        return False
    if u==False:
        return False
    return [mu,u]
```

This function returns a list of the multiplier and the first integral.

Example 3.

$$y'(y' + y^2x)' = 0 \to xy^2 + y' = C.$$

```
ode=diff(Y,x)*diff(diff(Y,x) + Y^2*x,x)
factorp(ode)
[1/diff(Y(x), x), x*Y(x)^2 + diff(Y(x), x)]
```

Example 4.

$$y'' + y^2 = 0 \to \frac{y^3}{3} + \frac{(y')^2}{2} = C.$$

```
ode=diff(Y,x,2) + Y^2
factorp(ode)
[diff(Y(x), x), 1/3*Y(x)^3 + 1/2*diff(Y(x), x)^2]
```

5. Conclusion

The problem of finding the integrating factor of an ordinary differential equation

$$f(x, y, y')y'' + g(x, y, y') = 0$$

has not been solved algorithmically in the general case, and there is little hope that this will change in the foreseeable future. For the case where $f = 1$, an algorithm for finding a factor depending on any two of the three variables x, y, y' was proposed in Ref. [4]. This algorithm, although quite complex, was implemented in Maple. In the article [5], it was proposed to return to this problem for an arbitrary f. In this article, we have shown that the factor can be found algorithmically under the assumption that it depends on one of the three variables x, y, y'. The text discusses in detail the case $\mu(y')$, which is most frequently encountered in practice. There are two problems that arise when implementing this algorithm. First, special cases arise during the solution process, which must be considered separately. Second, when calculating quadratures, integrals arise that are taken in elementary functions and even in rational ones, but modern integrators cannot recognize this.

References

[1] Malykh M. D. On integration in quadratures of the first-order ordinary differential equations // Mathematical Modelling and Geometry. — 2025. — Vol. 13, no. 2. — P. 16–31.

[2] Moses J. Symbolic Integration: The Stormy Decade // Communications of the ACM. — 1971. — Vol. 14, no. 8. — P. 548–560.

[3] Moses J. Symbolic integration. — Massachusetts institute of technology, 1967.

[4] Cheb-Terrab E. S., Roche A. D. Integrating factors for second order ODEs // Journal of Symbolic Computation. — 1999. — Vol. 27, no. 5. — P. 501–519.

[5] AlAhmad R., Al-Jararha M., Almefleh H. Exactness of Second Order Ordinary Differential Equations and Integrating Factors // Jordan Journal of Mathematics and Statistics. — 2015. — Vol. 8, no. 2. — P. 155–167.

[6] Carathéodory Constantin. Calculus of Variations and Partial Differential Equations of the First Order. — Holden-Day, 1965. — Vol. 1.

[7] Pavlov D. A. Symbolic integration // Computer tools in education. — 2010. — no. 5. — P. 38–43. — Access mode: `http://www.ipo.spb.ru/journal/index.php?article/1178`.

[8] Singer M.F. Liouvillian first integral of differential equations // Trans. Amer. Math. Soc. — 1992. — Vol. 333, no. 2. — P. 673–688.