Вычислительные технологии, 2022, том 27, № 6, с. 100–114. © ФИЦ ИВТ, 2022
Computational Technologies, 2022, vol. 27, no. 6, pp. 100–114. © FRC ICT, 2022

ISSN 1560-7534
eISSN 2313-691X

COMPUTATIONAL TECHNOLOGIES

DOI:10.25743/ICT.2022.27.6.009

Beam search for space extension in explicit ordinary differential equation conicalization

C. Gözükirmizi
Beykent University, 34485, İstanbul, Türkiye
Corresponding author: Gözükırmızı, Coşar, e-mail: cosargozukirmizi@beykent.edu.tr
Received April 22, 2022, revised May 19, 2022, accepted May 27, 2022.

Space extension for explicit ODEs is considering introduction of new equations to the equation set where the new unknowns are functionally dependent on the original unknowns. The purpose is to convert the ODE set into a form that has purely second degree multinomial right hand side functions. This is a necessary preprocessing step for certain series solution methods. Multinomial ODEs can be converted to ODEs with purely second degree terms through space extension. In a previous work, it is shown that the space extension with the smallest number of new unknowns can be found by a complete search. However, the complete search is not computationally efficient. In this paper, a computationally efficient search (beam search) is utilized but optimality (smallest number of new unknowns) is not guaranteed. The numerical experiments show that beam search is powerful in finding a useful space extension even for multinomials with relatively higher degrees.
Keywords: ordinary differential equations, space extension, beam search.
Citation: Gözükırmızı C. Beam search for space extension in explicit ordinary differential equation conicalization. Computational Technologies. 2022; 27(6):100–114. DOI:10.25743/ICT.2022.27.6.009.

Introduction

Explicit ordinary differential equations with multinomial right hand side functions appear in different fields. Although discretization methods dominate in many engineering applications, series expansion methods are also promising. Probabilistic evolution theory provides a series expansion method for ODE sets where the right hand side functions contain only second degree terms [1–11]. Article [9] is a detailed survey of probabilistic evolution theory.

To make the method applicable to a diverse set of problems, it is necessary to be able to convert an ODE set with multinomial right hand sides to an ODE set with purely second degree multinomial right hand sides. Conicalization means converting to second degree multinomial right hand side functions. This is done through space extension. For a long time, space extension was performed through trial and error to be able to transform the ODE into a asuitable form. Recently, it is shown that branch and bound search may be utilized to find the space extension with the smallest number of new function definitions (the optimal space extension) [10, 12]. However, such a complete search is not efficient. In this paper, beam search is proposed as a way to search for the optimal space extension. The implementation details and illustrative examples are also given in the following sections.

1. Space extension concept

Space extension is used in order to form new functions that are functionally dependent on the original functions but linearly independent from them. We form an ordinary differential equation on the new function and forget its functional dependence for a moment. This is done incrementally until the ODE set satisfies the desired characteristics. The new ODE set is larger than the original one, but represents exactly the same system. If the new ODE set is easier to solve, this means that the space extension served its purpose.

The concept is very useful for multinomial functions, but its use is not limited to multinomials. If the functions can form a closed set under a certain operator, then space extension can be directly used. Therefore, if the right hand sides also contain negative integer powers, exponentials or trigonometric functions in a certain way, the ideas here may be adapted to those cases also, but we will leave that for the future.

The functions whose derivatives appear on the left hand sides of the equations, can form a Hilbert space. The span of the Hilbert space is determined by the basis set formed by the functions whose derivatives appear on the left hand side. By appending new equations to the ODE set, it is always possible to go from an ODE set with multinomial right hand side functions to an ODE set with purely second degree multinomial right hand side functions. Appending an equation to the ODE set is performed through forming an ODE for a function that is linearly independent from the original left hand side functions. Therefore, appending an equation is also appending a new basis function to the Hilbert space, consequently "extending" the space.

1.1. Purely second degree right hand side functions

Our focus is on the original ODE sets with multinomial right hand side functions, and the purpose is to convert this set into a set with purely second degree right hand side functions. From this point on, by the term space extension, we will mean what is described in the previous sentence even though the space extension is a more general concept. More specifically, we will consider the case where the original ODE set consists of two equations. The reason for trying to obtain purely second degree right hand side functions is that, when we have such a structure, a series solution can easily be obtained. We will not get into the details of obtaining the series solution. It is explained in detail in different papers. Especially, [9] is a detailed survey on the subject.

All papers except [10] consider two consecutive steps for this task: space extension and constancy adding space extension. [10] showed this is not necessary because constancy adding space extension is also included in space extension. In order to explain why this is so, it is necessary to consider the constant function that has the value of 1 everywhere. We can call this function as $u^{(0,0)}$. This function is also the product of the zeroth power of one of the original functions and the zeroth power of the other one (considering that we start out with two unknowns). Therefore it is functionally dependent on the original functions. Then, we can forget the functional dependence for a moment and write an ODE for this function. Obviously, the right hand side will not have any terms. The right hand side will be the 0 function. Although 0 function does not seem to be a purely second degree term, it does not cause any problem (it is a purely second degree term with 0 as coefficient). We introduce this function to the ODE set when necessary. This means, we will introduce it when it appears on the right hand side of one of the ODEs. Therefore, what is performed for constancy adding space extension is exactly the same with what is performed for space extension.

1.2. Searching for space extension

Here, we will not repeat the concepts from [10], but it is important to emphasize the type of the search for the space extension. In [10], we proposed branch-and-bound search. This is a complete search. It is not possible to miss the optimal space extension. At each step all possible partitionings of the powers are analyzed, therefore nothing is left out. Also, people doing space extension by hand know that it is possible to make a space extension and end up with an ODE set worse than the original one (further away from being close to having purely second degree right hand side functions), and continue performing space extension infinitely without getting any result. This does not happen in branch-and-bound because branch-and-bound is a complete search. The score of the node will always take you to the optimal node. If the algorithm starts going in a bad direction, the score will suffer, and therefore the algorithm will continue with the better option. Of course, the admissible heuristic is the main component of how the algorithm determines what is better.

Instead of branch-and-bound search, breadth-first search can also be utilized. Although not powerful as branch-and-bound, breadth-first search does not need a score calculation and reordering at each step. On the other hand, depth-first search can cause problems. Depth-first search can lead to infinity problem described in the previous paragraph.

Although branch-and-bound search is very powerful and always gives the optimal space extension, it is not scalable. The number of necessary evaluations grows very fast when the powers of the right hand sides of the original ODEs grow. The tree becomes huge almost immediately. In this paper, we want to use a search that is easier to implement and requires less CPU and memory usage. In order to achieve that, we also make a compromise: the search does not need to always give the optimal solution. If it gives the optimal solution most of the time, and gives a solution that is very close to the optimal solution in the other times, it will be considered good enough.

1.3. Beam search

General information about beam search can be found in [13]. Beam search has been utilized with success in different domains including scheduling in supply chain models [14, 15], translation of text from one language to another [16, 17], speech recognition [18, 19] and computational biology [20].

In our application, we focus on beam search with branching factor w as 1. This simply means that at each step we will do the best space extension and prune the rest of the tree. Therefore we will be moving linearly, beaming our way to the space extension. The heuristic to be used is to always partition in half, because that is the fastest way to decrease the powers. If the number is even, partitioning in half is well-defined. If the number is odd, there are two options: the number $(2n-1)$ can be partitioned as $(n+(n-1))$ or $((n-1)+n)$. It is important to see this through an example.

Example 1.1. Assume that we have q^3p^3 as an additive right hand side term. This is a term with degree 6. Since the degree of the term is higher than 2, partitioning is necessary. There are eight ways to partition. They are as follows separated by the comma symbol: $(q^0p^0)\times(q^3p^3)$, $(q^0p^1)\times(q^3p^2)$, $(q^0p^2)\times(q^3p^1)$, $(q^0p^3)\times(q^3p^0)$, $(q^1p^0)\times(q^2p^3)$, $(q^1p^1)\times(q^2p^2)$, $(q^1p^2)\times(q^2p^1)$, $(q^1p^3)\times(q^2p^0)$. For any of these partitionings, if one momentarily forgets the meaning in the parenthesized expressions and considers each parenthesized expression as a function, all of these eight ways produce purely second degree terms. The heuristic advises us to partition in the middle. Due to commutativity of multiplication, we do not have second power of q on the left side of the product. For p, on the other hand, we do have the second power of p on the left side of the product. So, partitioning in the middle, we can have $(q^1p^1) \times (q^2p^2)$ or $(q^1p^2) \times (q^2p^1)$. We observed that $(q^1p^1) \times (q^2p^2)$ is a better choice. However, if we were partitioning q^1p^1, we would prefer $(q^0p^1) \times (q^1p^0)$ instead of $(q^0p^0) \times (q^1p^1)$ because we would already have (q^0p^1) and (q^1p^0), therefore, not causing the necessity of further space extension for these functions.

The heuristic is as follows. If the powers of the two functions are 1, avoid having the same power for q and p in the same function. Therefore q^1p^1 becomes $(q^0p^1) \times (q^1p^0)$. Otherwise, always make the power in the left multiplicand less than or equal to the corresponding one in the right multiplicand. To put it in other words, we have q^kp^ℓ such that $(k \neq 1) \vee (l \neq 1)$. Then the partitioning is $\left(q^{\lfloor k/2 \rfloor}p^{\lfloor \ell/2 \rfloor}\right) \times \left(q^{k-\lfloor k/2 \rfloor}p^{\ell-\lfloor \ell/2 \rfloor}\right)$.

2. Problem representation and solution

2.1. Representing the ODE set

We represent the ODE set by a vector. First, use the heuristic to partition the right hand sides (without space extension). Then fill the vector with the powers of the functions. Each sequence of 6 integers in the vector corresponds to a single right hand side term. Each pair within the 6 integers corresponds to the powers of the original functions. Of these 6 integers, the first pair has the powers for the left hand side, the middle pair has the powers for the first multiplicand of the right hand side term and the right pair has the powers for the second multiplicand of the right hand side term.

Explicit ODE with two unknowns may be represented as

$$\begin{gathered}\dot{x}(t) = \alpha_1 x(t)^{k_1} y(t)^{\ell_1} + \alpha_2 x(t)^{k_2} y(t)^{\ell_2} + \cdots + \alpha_m x(t)^{k_m} y(t)^{\ell_m},\\ \dot{y}(t) = \alpha_{m+1} x(t)^{k_{m+1}} y(t)^{\ell_{m+1}} + \alpha_{m+2} x(t)^{k_{m+2}} y(t)^{\ell_{m+2}} + \cdots + \alpha_{m+n} x(t)^{k_{m+n}} y(t)^{\ell_{m+n}},\\ \alpha_i \neq 0, \quad k_i, \ell_i \in \mathbb{N}, \quad i = 1, \ldots, (m+n),\end{gathered} \tag{1}$$

where the same (k_i, ℓ_i) pair of an additive term cannot appear more than once within the same equation. Let

$$u^{(k,l)} \equiv x(t)^k y(t)^\ell.$$

Then (1) can be written as

$$\begin{aligned}\dot{u}^{(1,0)} &= \alpha_1 u^{(k_1,\ell_1)} + \alpha_2 u^{(k_2,\ell_2)} + \cdots + \alpha_m u^{(k_m,\ell_m)},\\ \dot{u}^{(0,1)} &= \alpha_{m+1} u^{(k_{m+1},\ell_{m+1})} + \alpha_{m+2} u^{(k_{m+2},\ell_{m+2})} + \cdots + \alpha_{m+n} u^{(k_{m+n},\ell_{m+n})}\end{aligned}$$

and the pairs on the powers of the right hand side functions can be binary partitioned using the heuristic to form

$$\begin{aligned}\dot{u}^{(1,0)} &= \alpha_1 u^{(a_1,b_1)} u^{(c_1,d_1)} + \alpha_2 u^{(a_2,b_2)} u^{(c_2,d_2)} + \cdots + \alpha_m u^{(a_m,b_m)} u^{(c_m,d_m)},\\ \dot{u}^{(0,1)} &= \alpha_{m+1} u^{(a_{m+1},b_{m+1})} u^{(c_{m+1},d_{m+1})} + \alpha_{m+2} u^{(a_{m+2},b_{m+2})} u^{(c_{m+2},d_{m+2})} +\\ &\quad + \cdots + \alpha_{m+n} u^{(a_{m+n},b_{m+n})} u^{(c_{m+n},d_{m+n})}.\end{aligned} \tag{2}$$

The ODE in (2) can be represented by Table 1, where P stands for pair and Ind. stands for index. Each row has an index and a sextuple. Left pair is the pair that shows the left hand

T a b l e 1. ODE set before space extension

Ind.	ODE		
	Left P	Middle P	Right P
1	1, 0	a_1, b_1	c_1, d_1
2	1, 0	a_2, b_2	c_2, d_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$
m	1, 0	a_m, b_m	c_m, d_m
$m+1$	0, 1	a_{m+1}, b_{m+1}	c_{m+1}, d_{m+1}
$m+2$	0, 1	a_{m+2}, b_{m+2}	c_{m+2}, d_{m+2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$m+n$	0, 1	a_{m+n}, b_{m+n}	c_{m+n}, d_{m+n}

side. The middle pair and the right pair show the multiplicands of a single additive term on the right hand side.

The sextuples in each row of Table 1 can be put one by one into a vector. The vector that represents the ODE will then be [1, 0, a_1, b_1, c_1, d_1, 1, 0, a_2, b_2, c_2, d_2, …, 1, 0, a_m, b_m, c_m, d_m, 0, 1, a_{m+1}, b_{m+1}, c_{m+1}, d_{m+1}, 0, 1, a_{m+2}, b_{m+2}, c_{m+2}, d_{m+2}, …, 0, 1, a_{m+n}, b_{m+n}, c_{m+n}, d_{m+n}]. We also need another vector whose size is (1/6) of this vector. The vector is $[\alpha_1, \ldots, \alpha_m, \ldots, \alpha_{m+n}]$ and stores the coefficients.

Example 2.1. The van der Pol ODE is given as

$$\begin{aligned} \dot{x}(t) &= \mu x - \frac{\mu}{3}x^3 - \mu y, \quad x(0) = x_0, \\ \dot{y}(t) &= \frac{1}{\mu}x, \quad y(0) = y_0. \end{aligned} \tag{3}$$

Let $x(t)$ and $y(t)$ be the first and the second function respectively. Let

$$u^{(k,\ell)} \equiv x^k y^\ell$$

without showing the t-dependence explicitly for simplicity. Then, (3) can be rewritten as

$$\begin{aligned} \dot{u}^{(1,0)} &= \mu u^{(1,0)} - \frac{\mu}{3}u^{(3,0)} - \mu u^{(0,1)}, \\ \dot{u}^{(0,1)} &= \frac{1}{\mu}u^{(1,0)}. \end{aligned} \tag{4}$$

Using the heuristic, (4) becomes

$$\begin{aligned} \dot{u}^{(1,0)} &= \mu u^{(0,0)}u^{(1,0)} - \frac{\mu}{3}u^{(1,0)}u^{(2,0)} - \mu u^{(0,0)}u^{(0,1)}, \\ \dot{u}^{(0,1)} &= \frac{1}{\mu}u^{(0,0)}u^{(1,0)}, \end{aligned}$$

and for obtaining the vector representation of the ODE, the equations and the right hand side functions are taken into account in the order they appear. Then the vector representation of the van der Pol ODE is [1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 2, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0].

Example 2.2. The ODE set for the classical quartic anharmonic oscillator is given as

$$\begin{aligned} \dot{q}(t) &= \frac{1}{\mu}p(t), \quad q(0) = q_0, \\ \dot{p}(t) &= -k_1 q(t) - k_2 q(t)^3, \quad p(0) = p_0. \end{aligned} \tag{5}$$

Let $q(t)$ and $p(t)$ be the first and the second function respectively. Let

$$u^{(k,\ell)} \equiv q^k p^\ell$$

without showing the t-dependence explicitly for simplicity. Then, (5) can be rewritten as

$$\begin{aligned} \dot{u}^{(1,0)} &= \frac{1}{\mu}u^{(0,1)}, \\ \dot{u}^{(0,1)} &= -k_1 u^{(1,0)} - k_2 u^{(3,0)}. \end{aligned} \tag{6}$$

Using the heuristic, (6) becomes

$$\begin{aligned} \dot{u}^{(1,0)} &= \frac{1}{\mu}u^{(0,0)}u^{(0,1)}, \\ \dot{u}^{(0,1)} &= -k_1 u^{(0,0)}u^{(1,0)} - k_2 u^{(1,0)}u^{(2,0)} \end{aligned}$$

and for obtaining the vector representation of the ODE, the equations and the right hand side functions are taken into account in the order they appear. Then the vector representation for the classical quartic anharmonic oscillator is [1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0].

2.2. Extending the space

Starting out with one equation. The simplest case is to start with one ODE with constant right hand side function. The purpose is to obtain second degree multinomials on the right hand side through space extension. Let

$$\dot{u}^{(\ell)}(t) \equiv x(t)^\ell.$$

Then, ODE with constant right hand side is

$$\dot{u}^{(1)} = c,$$

which can also be written as

$$\dot{u}^{(1)} = cu^{(0)}u^{(0)}.$$

The space can be extended for $u^{(0)}$ giving

$$\dot{u}^{(0)} = 0.$$

Constant function 0 can be considered as a second degree multinomial with 0 as coefficient, because

$$\dot{u}^{(0)} = 0u^{(0)}u^{(0)}.$$

At this point, space extension is complete. All right hand side functions also appear on the left hand side.

The next simple case is to start with one ODE with polynomial right hand side function. The purpose is to obtain second degree multinomials on the right hand side through space extension. The ODE with polynomial (to be more precise, it is a monomial) right hand side is

$$\dot{u}^{(1)}(t) = u^{(\ell_1)},$$

where ℓ_1 is a positive integer. The first case to investigate is where $\ell_1 = 1$. Then, the equation is

$$\dot{u}^{(1)} = u^{(1)}.$$

Here, the right hand side is first degree. Since we need purely second degree right hand side we can rewrite as

$$\dot{u}^{(1)} = u^{(0)}u^{(1)},$$
$$\dot{u}^{(0)} = 0,$$

thus appending a new ODE to the set. The integer 1 is partitioned as $(0+1)$. After the operation, purely second degree structure is obtained. Although constant function 0 is not a second degree multinomial, it can be considered like a second degree multinomial with 0 as coefficient. Therefore, it is consistent with the structure.

The next case is where $\ell_1 = 2$. Therefore

$$\dot{u}^{(1)} = u^{(2)}$$

is under consideration. Then we can partition as $(0+2)$ or $(1+1)$. Partitioning as $(1+1)$ gives

$$\dot{u}^{(1)} = u^{(1)}u^{(1)}$$

and solves the problem by yielding second degree term on the right hand side.

The next case is where $\ell_1 = 3$. Then

$$\dot{u}^{(1)} = u^{(3)}$$

is under consideration. We can partition as $(0+3)$ or as $(1+2)$. We choose $(1+2)$ giving

$$\dot{u}^{(1)} = u^{(1)}u^{(2)},$$

where we also need to form an ODE for $u^{(2)}$. Using differentiation we can write the equation to append as

$$\dot{u}^{(2)} = 2u^{(1)}u^{(3)},$$

where $u^{(3)}$ also appears. We can repartition the right hand side of the above equation giving

$$\dot{u}^{(2)} = 2u^{(2)}u^{(2)}$$

providing purely second degree right hand side terms.

The next case is where $\ell_1 = 4$. Then

$$\dot{u}^{(1)} = u^{(4)}$$

is under consideration. Possible partitionings are $(0+4)$, $(1+3)$ and $(2+2)$. We choose $(2+2)$ giving

$$\dot{u}^{(1)} = u^{(2)}u^{(2)},$$

where we need to append an equation for $u^{(2)}$. Using differentiation

$$\dot{u}^{(2)} = 2u^{(1)}u^{(4)}$$

is obtained. We can repartition to get

$$\dot{u}^{(2)} = 2u^{(2)}u^{(3)} \tag{7}$$

therefore giving us the third power. We need to append an equation for $u^{(3)}$. It is

$$\dot{u}^{(3)} = 3u^{(2)}u^{(4)},$$

where the right hand side can be rewritten as

$$\dot{u}^{(3)} = 3u^{(3)}u^{(3)},$$

where the total power of five on the right hand side of (7) gave us two multiplicands of third power here. Also, purely second degree terms on the right hand side are also obtained.

Let us also exemplify a bad way to partition. Consider again

$$\dot{u}^{(1)} = u^{(4)}$$

and rewrite it as

$$\dot{u}^{(1)} = u^{(0)}u^{(4)},$$

where the right hand side is purely second degree. Now we need to append an equation for both $u^{(0)}$ and $u^{(4)}$. Thus, such partitioning does not provide any advantage. Appending the equation

$$\dot{u}^{(4)} = 4u^{(3)}u^{(4)} \tag{8}$$

we can observe that we have $u^{(3)}$ and $u^{(4)}$ on the right hand side. This is not a good partitioning considering that we also need to append an equation for $u^{(0)}$ and we have appended an equation for $u^{(4)}$ without the need for it.

Also, from (8), we can continue stubbornly to rewrite as

$$\dot{u}^{(4)} = 4u^{(0)}u^{(7)}$$

and then try to append an equation for $u^{(7)}$. The problem of partitioning $u^{(4)}$ has become the problem of partitioning $u^{(7)}$. The form has become more distant to having purely second degree multinomial right hand sides.

Starting out with two equations. Let $x(t)$ and $y(t)$ be the first and the second function respectively. Also let

$$u^{(\ell_1,\ell_2)}(t) \equiv x(t)^{\ell_1} y(t)^{\ell_2}, \quad \ell_1, \ell_2 \in \mathbb{N}.$$

Then, the general form of the original ODE set is

$$\dot{u}^{(1,0)}(t) = \sum_{\ell_1=0}^{\infty} \sum_{\ell_2=0}^{\infty} a_1^{(\ell_1,\ell_2)} u^{(\ell_1,\ell_2)}(t),$$
$$\dot{u}^{(0,1)}(t) = \sum_{\ell_1=0}^{\infty} \sum_{\ell_2=0}^{\infty} a_2^{(\ell_1,\ell_2)} u^{(\ell_1,\ell_2)}(t).$$

The expectation is that only a small number of a coefficients on the right hand side are nonzero and the others are zero. We have here multinomial right hand side functions. Using the heuristic, each additive term on the right hand side will be written as the product of two multinomials. If these multinomials do not exist on the left hand side, a new ODE will be obtained for them using

$$\dot{u}^{(\ell_1,\ell_2)}(t) = \ell_1 u^{(\ell_1-1,\ell_2)}(t) \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{\infty} a_1^{(j_1,j_2)} u^{(j_1,j_2)}(t) + \ell_2 u^{(\ell_1,\ell_2-1)}(t) \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{\infty} a_2^{(j_1,j_2)} u^{(j_1,j_2)}(t) \quad (9)$$

and appended to the set. The process will be continued for all of the additive terms for all of the equations in the set. At some point, all of the multinomials on the right hand side will also appear on the left hand side and the space extension will be complete.

The right hand side of the equation is the zero function when the left hand side is a constant. This corresponds to constancy adding space extension. For this situation, the pairs for the right hand side should not contain anything because there is nothing to partition. As a notation convention, we show these right hand side pairs with all 0 values.

Example 2.3. The ODE set for van der Pol ODE is [1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 2, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0]. The space extension is shown by Table 2. In the table P. stands for pair. The first four rows of the table is the original ODE set. The other rows are filled in incrementally. The right hand sides (middle pairs and right pairs) are analyzed one by one. The first right hand side pair (first row middle pair) is 0, 0. It does not exist as left pair, therefore space should be extended for it. The fifth row is written adding constancy to the ODE set. The right pair on the first row already exists as left pair. The middle pair on second row already exists as left pair. The right pair on second row does not exist as left pair, therefore space should be extended for it. Observe (9). The middle pairs and right pairs of the first four rows take part in all space extensions. Using the heuristic, repartitioning will be performed. Observe row 6. We know that we want to write an equation for 2, 0. Therefore that is our left pair. Then decrement the first number of the left pair by 1 and add it to the sum of the middle pair and the right pair of row 1. The result would be 2, 0. Using the heuristic, partition it and write the middle pair and right pair of row 6. Then, go to row 7 and write your left pair. It is still 2, 0 because we have not finished working on it. Decrement the first number of the left pair of row 7 by 1 and add it to the sum of the middle pair and the right pair of row 2. The result would be 4, 0. Using the heuristic, partition it and write the middle pair and right pair of row 7. Then, go to row 8 and write your left pair. It is still 2, 0 because we have not finished working on it. Decrement the first number of the left pair of row 8 by 1 and add it to the sum of the middle pair and the right pair of row 3. The result would be 1, 1. Using the heuristic, partition it and write the middle pair and right pair of row 8. Then we need to decrement the right number of 2, 0. However, it is already 0 and therefore cannot be decremented. So, the equation for 2, 0 is complete. Also, now, all the middle pairs and right pairs also exist as a left pair. Therefore, the space extension is complete. There are 4 unique left pairs which means that the new ODE set has 4 equations in total. This space extension is also the optimal space extension. The branch-and-bound search also yields this particular space extension with 4 equations.

Example 2.4. The ODE set for classical quartic anharmonic oscillator is [1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0]. The space extension is shown by Table 3. In the table P stands

T a b l e 2. Space extension of van der Pol ODE

Ind.	ODE		
	Left P	Middle P	Right P
1	1, 0	0, 0	1, 0
2	1, 0	1, 0	2, 0
3	1, 0	0, 0	0, 1
4	0, 1	0, 0	1, 0
5	0, 0	0, 0	0, 0
6	2, 0	1, 0	1, 0
7	2, 0	2, 0	2, 0
8	2, 0	0, 1	1, 0

T a b l e 3. Space extension of classical quartic anharmonic oscillator

Ind.	ODE		
	Left P	Middle P	Right P
1	1, 0	0, 0	0, 1
2	0, 1	0, 0	1, 0
3	0, 1	1, 0	2, 0
4	0, 0	0, 0	0, 0
5	2, 0	0, 1	1, 0

for pair. The first four rows of the table is the original ODE set. The other rows are filled in incrementally. The right hand sides (middle pairs and right pairs) are analyzed one by one. The first right hand side pair (first row middle pair) is 0, 0. It does not exist as left pair, therefore space should be extended for it. The fourth row is written adding constancy to the ODE set. The right pair on the first row already exists as left pair. The middle pair on second row already exists as left pair. The right pair on second row already exists left pair. The middle pair on third row already exists as left pair. The right pair on third row does not exist as left pair, therefore space should be extended for it. Observe (9). The middle pairs and right pairs of the first three rows take part in all space extensions. Using the heuristic, repartitioning will be performed. Observe row 5. We know that we want to write an equation for 2, 0. Therefore that is our left pair. Then decrement the first number of the left pair by 1 and add it to the sum of the middle pair and the right pair of row 1. The result would be 1, 1. Using the heuristic, partition it and write the middle pair and right pair of row 5. Then we need to decrement the right number of 2, 0. However, it is already 0 and therefore cannot be decremented. So, the equation for 2, 0 is complete. Also, now, all the middle pairs and right pairs also exist as a left pair. Therefore, the space extension is complete. There are 4 unique left pairs which means that the new ODE set has 4 equations in total. This space extension is also the optimal space extension. The branch-and-bound search also yields this particular space extension with 4 equations.

Example 2.5. Let us consider the ODE set given by [1, 0, 50, 50, 50, 50, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0]. Here we have very high powers on the right hand side of the first equation. Let $x(t)$ and $y(t)$ be the first and the second function respectively. Let

$$u^{(k,\ell)} \equiv x^k y^\ell.$$

Then, the equation set represented by the vector is

$$\dot{u}^{(1,0)} = c_1 u^{(50,50)} u^{(50,50)},$$
$$\dot{u}^{(0,1)} = c_2 u^{(0,0)} u^{(1,0)} + c_3 u^{(1,0)} u^{(2,0)},$$

where c coefficients are nonzero scalars. Using beam search, purely second degree right hand sides were obtained with an ODE set of 303 equations. In this ODE set, there are 898 additive right hand side terms in total. Table 4 shows the first 11 of these 898 right hand side terms of the new ODE set. We do not know if this is the optimal space extension.

Table 4. Space extension of [1, 0, 50, 50, 50, 50, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0]

Ind.	ODE		
	Left P	Middle P	Right P
1	1, 0	50, 50	50, 50
2	0, 1	0, 0	1, 0
3	0, 1	1, 0	2, 0
4	50, 50	74, 75	75, 75
5	50, 50	25, 24	26, 25
6	50, 50	26, 24	27, 25
7	0, 0	0, 0	0, 0
8	2, 0	50, 50	51, 50
9	74, 75	86, 87	87, 88
10	74, 75	37, 37	38, 37
11	74, 75	38, 37	39, 37
$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 5. Total number of equations after space extension for van der Pol (Example 2.3), classical quartic anharmonic oscillator (Example 2.4) and [1, 0, 50, 50, 50, 50, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0] (Example 2.5). H1, H2 and H3 are Heuristic 1, Heuristic 2 and Heuristic 3 respectively

Equation	Heuristic		
	H1	H2	H3
Van der Pol	4	6	4
Anharmonic oscillator	4	5	4
High powers	303	303	491

We have also tried different heuristics that also satisfy the idea of partitioning in the middle. In the aforementioned examples, the heuristic is

$$q^1p^1 = (q^0p^1) \times (q^1p^0),$$
$$q^kp^\ell = \left(q^{\lfloor k/2\rfloor}p^{\lfloor \ell/2\rfloor}\right) \times \left(q^{k-\lfloor k/2\rfloor}p^{\ell-\lfloor \ell/2\rfloor}\right), \quad (k \neq 1) \vee (\ell \neq 1).$$

Let us name the heuristic above as Heuristic 1 (or H1). Also, let Heuristic 2 (or H2) be

$$q^kp^\ell = \left(q^{\lfloor k/2\rfloor}p^{\lfloor \ell/2\rfloor}\right) \times \left(q^{k-\lfloor k/2\rfloor}p^{\ell-\lfloor \ell/2\rfloor}\right), \quad k = 0,1,2,\ldots, \quad \ell = 0,1,2,\ldots$$

and let Heuristic 3 (or H3) be

$$q^kp^\ell = \left(q^{\lfloor k/2\rfloor}p^{\lceil \ell/2\rceil}\right) \times \left(q^{k-\lfloor k/2\rfloor}p^{\ell-\lceil \ell/2\rceil}\right), \quad k = 0,1,2,\ldots, \quad \ell = 0,1,2,\ldots.$$

These three heuristics were utilized with the equation sets in examples 2.3–2.5. The total number of equations are shown in Table 5.

Table 5 shows that the heuristic matters. Heuristic 1 is the best heuristic for these examples. In general, Heuristic 1 does not always give the optimal space extension. On the other hand, it can be shown that Heuristic 1 always gives a valid conicalization (not necessarily optimal space extension). Also, it is important to mention that it is possible to increase the chance of finding the optimal space extension by using beam search where the search parameter $w > 1$. That means, looking at more parts of the tree by possibly taking more than one path from each node.

2.3. Implementation

For implementation, C++ programming language is used. The ODE set is represented through the vector data structure. Space extension is modeled through the arithmetic of the integers within the vector of integers. Space extension corresponds to augmentation of the vector of integers. Each sextuple in the vector represents a single additive term. Within each sextuple, left pair is the left hand side of the equation, whereas the middle pair and the right pair are the multiplicands within the term at the right hand side.

2.4. Starting out with more than two equations

The approach can be generalized to start with more than two equations. To start with n equations and n unknowns, the representation is as follows

$$u^{(\ell_1,\ell_2,\ldots,\ell_n)}(t) \equiv x_1(t)^{\ell_1} x_2(t)^{\ell_2} \ldots x_n(t)^{\ell_n}, \quad \ell_1, \ell_2, \ldots, \ell_n \in \mathbb{N},$$

where whole set will be represented through the functions above. The tuple representation is given in Table 6.

The heuristic for this multidimensional case is given as follows. Again, we will distinguish the situation where there are 1s and where there are not. In the beginning also, the set should be rewritten (integers should be repartitioned) according to the heuristic. If the powers do not include 1, the heuristic is

$$x_1^{p_1} x_2^{p_2} \ldots x_n^{p_n} = \left(x_1^{\lfloor p_1/2\rfloor} x_2^{\lfloor p_2/2\rfloor} \ldots x_n^{\lfloor p_n/2\rfloor}\right) \times \left(x_1^{p_1-\lfloor p_1/2\rfloor} x_2^{p_2-\lfloor p_2/2\rfloor} \ldots x_n^{p_n-\lfloor p_n/2\rfloor}\right),$$
$$p_j \neq 1, \quad j = 1, \ldots, n$$

and if there are the values 1, the situation is different. We need to count the occurrences of 1s from left to right. If it is the first, third, etc. occurrence, then we perform floor operation. If it is the second, fourth, etc. occurrence, then we perform the ceiling operation. This heuristic is also consistent with what we have done for the two-dimensional case. Therefore,

$$x_1^{p_1} x_2^{p_2} \ldots x_n^{p_n} = \left(x_1^{\mathcal{L}(p_1/2)} x_2^{\mathcal{L}(p_2/2)} \ldots x_n^{\mathcal{L}(p_n/2)}\right) \times \left(x_1^{p_1-\mathcal{L}(p_1/2)} x_2^{p_2-\mathcal{L}(p_2/2)} \ldots x_n^{p_n-\mathcal{L}(p_n/2)}\right) \quad (10)$$

is used where the operator is defined as

$$\mathcal{L}(p_k/2) = \lfloor p_k/2\rfloor,$$
$$(p_k \neq 1) \vee ((p_k = 1) \wedge \text{it is the occurrence } 1, 3, 5, \ldots),$$
$$k = 1, 2, \ldots, n,$$
$$\mathcal{L}(p_k/2) = \lceil p_k/2\rceil,$$
$$((p_k = 1) \wedge \text{it is the occurrence } 2, 4, 6, \ldots),$$
$$k = 1, 2, \ldots, n.$$

As a notation convention, if the left hand side tuple representation is all 0s, the two right hand side tuple representations will also be all 0s. Actually, all 0s in a tuple means a nonzero constant function, but for the constancy adding space extension, the zero constant function is also represented in the same way.

Example 2.6. The Henon – Heiles ODE is an explicit ODE with multinomial right hand sides. It is given by

$$\dot{x} = p_x,$$
$$\dot{p}_x = -x - 2\lambda xy,$$
$$\dot{y} = p_y,$$
$$\dot{p}_y = -y - \lambda x^2 + \lambda y^2$$

T a b l e 6. Representation of multinomials

Tuple representation	Function representation
$p_1, p_2, \ldots, p_{n-1}, p_n$	$x_1^{p_1} x_2^{p_2} \ldots x_{n-1}^{p_{n-1}} x_n^{p_n}$

Table 7. Space extension of Henon – Heiles ODE

Ind.	ODE		
	Left Q	Middle Q	Right Q
1	1, 0, 0, 0	0, 0, 0, 0	0, 1, 0, 0
2	0, 1, 0, 0	0, 0, 0, 0	1, 0, 0, 0
3	0, 1, 0, 0	0, 0, 1, 0	1, 0, 0, 0
4	0, 0, 1, 0	0, 0, 0, 0	0, 0, 0, 1
5	0, 0, 0, 1	0, 0, 0, 0	0, 0, 1, 0
6	0, 0, 0, 1	1, 0, 0, 0	1, 0, 0, 0
7	0, 0, 0, 1	0, 0, 1, 0	0, 0, 1, 0
8	0, 0, 0, 0	0, 0, 0, 0	0, 0, 0, 0

consisting of four equations and four unknowns [11]. Now, each function will be a quadruple (a sequence of four integers). Each additive term will be represented with three quadruples: a quadruple for the left hand side, a quadruple for the first multiplicand of the right hand side term and a quadruple for the second multiplicand of the right hand side term. Let

$$u^{(k,\ell,m,n)} \equiv x^k p_x^\ell y^m p_y^n$$

then the equation can be rewritten as

$$\begin{aligned}
\dot{u}^{(1,0,0,0)} &= u^{(0,0,0,0)} u^{(0,1,0,0)}, \\
\dot{u}^{(0,1,0,0)} &= u^{(0,0,0,0)} u^{(1,0,0,0)} + u^{(0,0,1,0)}\ u^{(1,0,0,0)}, \\
\dot{u}^{(0,0,1,0)} &= u^{(0,0,0,0)} u^{(0,0,0,1)}, \\
\dot{u}^{(0,0,0,1)} &= u^{(0,0,0,0)} u^{(0,0,1,0)} + u^{(1,0,0,0)} u^{(1,0,0,0)} + u^{(0,0,1,0)} u^{(0,0,1,0)}
\end{aligned}$$

using the heuristic in (10). The equation set is shown in the first seven rows of Table 7. The next step is to look at the middle and right quadruples row by row. If the quadruple does not exist as a left quadruple, space should be extended, adding new rows to the table. The middle quadruple of the first row does not exist as a left quadruple, therefore space should be extended for it. Since it is all 0s, it is the constancy adding space extension. Therefore a row with all 0s is appended to the table to form the eighth row of the table. The other middle quadruples and right quadruples exist as a left quadruple, therefore space extension is complete. This space extension is also the optimal space extension.

Concluding remarks

Beam search was utilized for obtaining an ODE set with purely second degree right hand side functions. In order to improve the chance of hitting a better space extension by covering more ground, it is possible to increase the beam search parameter from 1 to a higher integer. That means, at each node, instead of choosing the best path, choose the best w paths, and take all of the w paths into account.

Acknowledgements. The author would like to thank Metin Demiralp for fruitful discussions during the preparation of the manuscript.

References

[1] **Gözükırmızı C., Demiralp M.** Probabilistic evolution theory for explicit autonomous ordinary differential equations: recursion of squarified telescope matrices and optimal space extension. Journal of Mathematical Chemistry. 2018; 56(7):1826–1848. DOI:10.1007/s10910-017-0849-2.

[2] **Gözükırmızı C., Kırkın M.E.** Classical symmetric fourth degree potential systems in probabilistic evolution theoretical perspective: most facilitative conicalization and squarification of telescope matrices. AIP Conference Proceedings. 2017; 1798(1):020061. DOI:10.1063/1.4972653.

[3] **Gözükırmızı C., Tataroğlu E.** Squarification of telescope matrices in the probabilistic evolution theoretical approach to the two particle classical mechanics as an illustrative implementation. AIP Conference Proceedings. 2017; 1798(1):020062. DOI:10.1063/1.4972654.

[4] **Kırkın M.E., Gözükırmızı C.** Probabilistic evolution theory for ode sets with second degree multinomial right hand side functions: certain reductive cases. AIP Conference Proceedings. 2015; 1702(1):170012. DOI:10.1063/1.4938947.

[5] **Gözükırmızı C., Demiralp M.** Probabilistic evolution theory for ode sets with second degree multinomial right hand side functions: implementation. AIP Conference Proceedings. 2015; 1702(1):170002. DOI:10.1063/1.4938937.

[6] **Gözükırmızı C., Demiralp M.** Constancy adding space extension for ode sets with second degree multinomial right hand side functions. AIP Conference Proceedings. 2014; 1618(1):875–878. DOI:10.1063/1.4897872.

[7] **Gözükırmızı C., Demiralp M.** Probabilistic evolution approach for the solution of explicit autonomous ordinary differential equations. Part 1: Arbitrariness and equipartition theorem in kronecker power series. Journal of Mathematical Chemistry. 2014; 52(3):866–880. DOI:10.1007/s10910-013-0298-5.

[8] **Gözükırmızı C., Demiralp M.** Probabilistic evolution approach for the solution of explicit autonomous ordinary differential equations. Part 2: Kernel separability, space extension, and, series solution via telescopic matrices. Journal of Mathematical Chemistry. 2014; 52(3):881–898. DOI:10.1007/s10910-013-0299-4.

[9] **Demiralp M.** Probabilistic evolution theory in its basic aspects, and, its fundamental progressive stages. Mathematics in Engineering, Science, Aerospace (MESA). 2018; 9(2):245–275.

[10] **Gözükırmızı C., Demiralp M.** Solving odes by obtaining purely second degree multinomials via branch and bound with admissible heuristic. Mathematics. 2019; 7(4):367. DOI:10.3390/math7040367.

[11] **Gözükırmızı C., Kırkın M.E., Demiralp M.** Probabilistic evolution theory for the solution of explicit autonomous ordinary differential equations: squarified telescope matrices. Journal of Mathematical Chemistry. 2017; 55(1):175–194. DOI:10.1007/s10910-016-0678-8.

[12] **Bychkov A., Pogudin G.** Optimal monomial quadratization for ODE systems. Flocchini P., Moura L. (Eds). Combinatorial algorithms. Springer International Publishing; 2021: 122–136. DOI:10.1007/978-3-030-79987-8_9.

[13] **Winston P.H.** Artificial intelligence (3rd ed.). USA: Addison-Wesley Longman Publishing Co.; 1992: 737. DOI:10.5555/129914.

[14] **Habenicht K., Monch L.** A finite-capacity beam-search-algorithm for production scheduling in semiconductor manufacturing. Proceedings of the Winter Simulation Conference. 2002; (2):1406–1413. DOI:10.1109/WSC.2002.1166411.

[15] **Ow P.S., Morton T.E.** Filtered beam search in scheduling. International Journal of Production Research. 1988; 26(1):35–62. DOI:10.1080/00207548808947840.

[16] **Koehn P.** Pharaoh: a beam search decoder for phrase-based statistical machine translation models. Frederking R.E., Taylor K.B. (eds) Machine translation: from real users to research. Berlin, Heidelberg; Springer Berlin Heidelberg; 2004: 115–124. DOI:10.1007/978-3-540-30194-3_13.

[17] **Tillmann C., Ney H.** Word reordering and a dynamic programming beam search algorithm for statistical machine translation. Computational Linguistics. 2003; 29(1):97–133. DOI:10.1162/089120103321337458.

[18] **Ney H., Haeb-Umbach R., Tran B.-H., Oerder M.** Improvements in beam search for 10000-word continuous speech recognition. 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing. 1992; (1):9–12. DOI:10.1109/ICASSP.1992.225985.

[19] **Ortmanns S., Ney H.** Look-ahead techniques for fast beam search. Computer Speech and Language. 2000; 14(1):15–32. DOI:10.1006/csla.1999.0131.

[20] **Blum C., Blesa M.J., López-Ibáñez M.** Beam search for the longest common subsequence problem. Computers and Operations Research. 2009; 36(12):3178–3186. DOI:10.1016/j.cor.2009.02.005.

Вычислительные технологии, 2022, том 27, № 6, с. 100–114. © ФИЦ ИВТ, 2022 ISSN 1560-7534
Computational Technologies, 2022, vol. 27, no. 6, pp. 100–114. © FRC ICT, 2022 eISSN 2313-691X

ВЫЧИСЛИТЕЛЬНЫЕ ТЕХНОЛОГИИ

DOI:10.25743/ICT.2022.27.6.009

Лучевой поиск для расширения пространства обыкновенных дифференциальных уравнений, разрешенных относительно производной

Д. Гёзюкырмызы

Департамент компьютерной инженерии, Университет Бейкент, 34485, Стамбул, Турция
Контактный автор: Гёзюкырмызы Джёшар, e-mail: cosargozukirmizi@beykent.edu.tr
Поступила 22 апреля 2022 г., доработана 19 мая 2022 г. принята в печать 27 мая 2022 г.

Аннотация

Расширение пространства для явных ОДУ заключается во введении новых уравнений в набор уравнений, где новые неизвестные функционально зависят от исходных неизвестных. Цель состоит в том, чтобы преобразовать набор ОДУ в форму, которая имеет чисто полиномиальные правые части второй степени. Это необходимый шаг предварительной обработки для ряда некоторых методов решения. Полиномиальные ОДУ могут быть преобразованы в ОДУ с членами чисто второй степени посредством пространственного расширения. В предыдущей работе показано, что расширение пространства с наименьшим числом новых неизвестных может быть найдено полным перебором. Однако полный поиск неэффективен в вычислительном отношении. В этой статье используется эффективный в вычислительном отношении поиск (поиск луча), но оптимальность (наименьшее количество новых неизвестных) не гарантируется. Численные эксперименты показывают, что поиск луча помогает найти полезное расширение пространства даже для многочленов с относительно более высокими степенями.

Ключевые слова: обыкновенные дифференциальные уравнения, расширение пространства, поиск луча.

Цитирование: Гёзюкырмызы Д. Лучевой поиск для расширения пространства обыкновенных дифференциальных уравнений, разрешенных относительно производной. Вычислительные технологии. 2022; 27(6):100–114. DOI:10.25743/ICT.2022.27.6.009. (на английском)

Благодарности. Автор хотел бы поблагодарить Метина Демиралпа за плодотворные обсуждения во время подготовки рукописи.