A NEW METHOD FOR FUNCTIONAL DECOMPOSITION OF RATIONAL INVARIANTS, AND THE SOLUTION OF ABEL'S DIFFERENTIAL EQUATION VIA THE EQUIVALENCE METHOD

by

Austin Duncan Roche

B.Math., University of Waterloo, 1999

M.Sc., McGill University, 2001

a Thesis submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
in the Department
of
Mathematics

SIMON FRASER UNIVERSITY
Summer 2010

APPROVAL

Name: Austin Duncan Roche

Degree: Doctor of Philosophy

Title of Thesis: A New Method for Functional Decomposition of Rational Invariants, and the Solution of Abel's Differential Equation via the Equivalence Method

Examining Committee: Dr. Vahid Dabbaghian
Chair

Dr. Peter Borwein, Senior Supervisor

Dr. Edgardo Cheb-Terrab, Co-Supervisor,
Research Fellow
Maplesoft, Waterloo Maple Inc.

Dr. Michael Monagan, Supervisor

Dr. Petr Lisonek, SFU Examiner

Dr. Robert Corless, External Examiner,
Professor of Mathematics,
University of Western Ontario

Date Approved: 3 August 2010

SIMON FRASER UNIVERSITY
LIBRARY

Declaration of Partial Copyright Licence

Abstract

The equivalence method for ordinary differential equations (ODEs) involves finding a transformation mapping a given equation into a target equation with known solution. Such equivalence transformations are found by solving systems relating the differential invariants of the input equation to those of the target equation. Standard solution techniques, applicable when the invariants are rational functions, use algebraic elimination and can solve complete families of equations characterized by many parameters. However, the complexity of such methods increases exponentially with the number of parameters, and can become impractical in interesting cases.

A new technique is presented for overdetermined rational function decomposition, specifically tailored to systems of invariants. Its complexity is effectively independent of the number of parameters defining the system. The use of this technique in the ODE equivalence method is described, including the use of minimal invariants which ensures the solution of all rational coefficient equations in the target class. Recognizing that related invariants tend to be composed as products of powers of a set of common polynomials, and furthermore that the pattern of these polynomials is invariant under composition by rational functions, we can compute these component polynomials invariantly. Using the target system as a model, a sequence of invariant computations is built that successively simplifies the system, leading eventually to the determination of the parameters and transformation function. The resulting algorithm mimics a formula in its specificity but lacks the associated expression growth. Additionally, necessary conditions are checked after each step, minimizing time spent testing invalid classes.

For certain parameter combinations, the structure of the component polynomials changes and the general algorithm can fail. Such cases are analysed in advance and a hierarchy of sub-algorithms is built to handle them, resulting in one super-algorithm to match the full super-class.

The new equivalence method is demonstrated by the implementation in Maple of a first complete algorithm for the solution of Abel differential equations of the inverse-linear or inverse-Riccati classes. Together these two super-classes, depending on two and three

parameters respectively, comprise the bulk of the solvable Abel classes described in the literature.

To Max, Dorothy, Jenny and Sergei
with love

Acknowledgments

This research was supported in part by the Centre for Experimental and Constructive Mathematics, funded by MITACS (Mathematics of Information Technology and Complex Systems); and a research assistantship provided by Peter Borwein, funded by the Natural Sciences and Engineering Research Council of Canada. Thank you Peter for bearing with me.

Thanks especially to Edgardo Cheb-Terrab, who introduced the subject and has been the guide for the work. You have been a mentor and a friend and I am honoured to have had this opportunity.

I have also had useful discussions with Michael Monagan, Theodore Kolokolnikov and Greg Reid and I thank them for their advice.

Thank you to Maplesoft for giving me a job and allowing me enough time to get this thesis finished.

Thank you Erik Postma for your encouragement and technical expertise.

Especially, thank you to my family, Mum, Dad, Geoff, Ash, and Nanny for your patience, love and support; and a special thank you to Natalie, Jacqueline, Michelle and Ava.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The equivalence method for obtaining exact solutions for differential equations consists in matching a given input equation, via an invertible transformation that can be computed, to an equivalent target equation among a collection of equations with known solutions. The solution to the input equation is then obtained by applying the inverse transformation to the solution of the target equation.

Given target and input equations $\Omega = 0$ and $\tilde{\Omega} = 0$ respectively, the equivalence problem can be formulated as an equation $\mu \tilde{\Omega} = \Omega \circ \tau$ to be solved for the transformation function τ and an integrating factor μ. If the target and input equations have a specific polynomial form (and this is preserved by τ) we can express μ in terms of τ and equate coefficients, obtaining a system for the transformation functions and their derivatives. When this system is overdetermined we can eliminate derivatives, reducing further to an algebraic system $\{\tilde{I}_i = I_i \circ \tau\}$. This system can be solved as a functional decomposition; the functions $\tilde{I}_i$ and I_i are known as invariants. In many important problems these invariants are rational [8, 9] and can be tackled by a standard algebraic elimination using resultants and greatest common divisors. To the best of our knowledge this approach was first suggested by Liouville [16]; it has been implemented successfully in [8] for example.

A relevant generalization is to simultaneously match the input equation to the infinite set of classes defined by a parametric target equation [9]. This case is more difficult because the parameters must be determined as well as the equivalence transformation. The natural approach, extending the algebraic elimination to include these extra unknowns, requires an additional invariant for each extra parameter. In practice this method can become intractable due to expression swell, even with just two constant parameters defining the class [8].

In this work we introduce a new technique which exploits the special structure that the invariants have by their nature as the result of a differential elimination: that they tend to be composed as products of many common polynomial subexpressions, often raised to large integer powers. Analyzing this structure has two benefits. Firstly, it reduces the known expressions in the decomposition from a small set of large, high degree rational expressions to a larger set of smaller, low degree polynomials. It also produces a characteristic numerical signature, invariant under rational transformations, which can be computed at minimal cost and used to significantly restrict the search space of target classes. The combination of these two simplifications is so effective that the bottleneck in the algorithm becomes computation of the invariants themselves: the time required for the remaining steps, including the determination of the unknown parameters, is comparatively insignificant. The complexity of the resulting matching algorithm is therefore effectively independent of the number of parameters defining the class. The method, to be described in the following, is sufficient for example to provide an efficient solving algorithm for both the 2-parameter Abel inverse-linear and 3-parameter Abel inverse-Riccati problems [9], which to the best of our knowledge are intractable using other known approaches.

In section 2.1 we will demonstrate the process of converting the equivalence problem to an algebraic system in terms of the invariants. We will then use this derivation to show the inherent structure that invariants have. In Section 2.2 we will describe a new approach to the algebraic elimination subproblem, based fundamentally on using this structure, that breaks down the invariants into a set of component polynomials. We also describe the concept of minimal forms for the target equation and corresponding invariants, which are necessary for the new approach to work. Finally, in Section 2.4 we describe how to use this new approach effectively to solve parameterized functional decomposition problems. Throughout we will demonstrate the various concepts by means of examples coming from the problem of Abel equations. We summarize the complete approach in Section 2.5.

In Chapter 3 we demonstrate the use of these techniques in the development of a first complete solution algorithm for the two-parameter Abel Inverse Linear and three-parameter Abel Inverse Riccati classes of ordinary differential equations. Together these two superclasses, depending on two and three parameters respectively, comprise the bulk of the solvable Abel classes described in the literature [1, 5, 7–9, 12–18, 21–23, 27].

Chapter 2

Functional Decomposition of Rational Invariants

2.1 Differential equations, equivalence and invariants

In this section we shall define the equivalence problem, restrict it to a case which is feasible to solve, and show how classical methods reduce the problem to an algebraic system involving the decomposition of characteristic functions called invariants. We will then describe the structural properties of invariants that suggest a new approach for solving this decomposition.

Consider a family of ordinary differential equations, $\mathcal{F} = \{\Omega_i(x, y, y', \ldots) = 0\}$,[1] a subset $\mathcal{F}_s$ of equations known to be solvable, and a set $\mathcal{G}$ of invertible point transformations acting on $\mathcal{F}$,[2]

$$\tau : (x, y) \mapsto (\chi(x, y), \psi(x, y))\,, \tag{2.1}$$

where χ and ψ are arbitrary functions parameterizing $\mathcal{G}$. Our aim is to develop a matching algorithm which takes any input equation $\tilde{\Omega} \in \mathcal{F}$, and computes, when they exist, a target equation $\Omega \in \mathcal{F}_s$, a transformation $\tau \in \mathcal{G}$ and an integrating factor μ not depending on the highest derivative in Ω, such that $\tilde{\Omega} = \mu\,\Omega \circ \tau$, that is:[3]

$$\tilde{\Omega}(x, y, y', \ldots, y^{(n)}) = \mu(x, y, \ldots, y^{(n-1)})\,\Omega\left(\chi(x, y), \psi(x, y), \frac{D\psi}{D\chi}, \ldots, \frac{D^{(n)}\psi}{D^{(n)}\chi}\right). \tag{2.2}$$

---

[1] Throughout this work the symbol $'$ will denote a derivative.

[2] We abuse notation somewhat here in that we consider an ODE Ω to be a member of the family $\mathcal{F}$ when there is a non-zero integrating factor μ not depending on the highest derivative that occurs in Ω such that $\mu\,\Omega \in \mathcal{F}$.

[3] D represents the total derivative, ie. $D\psi = \psi_{\tilde{x}} + \psi_{\tilde{y}}\tilde{y}'$.

In this work we consider the case where the family $\mathcal{F}$ consists of ODEs with a specific polynomial form in the dependent variable y and its derivatives, which we call *poly-y*. The coefficients of y and its derivatives can depend on arbitrary functions of x. For example, Abel's differential equation, given by[4]

$$\Omega(x, y, y') \equiv (g_1 y + g_0)y' - (f_3 y^3 + f_2 y^2 + f_1 y + f_0) = 0, \tag{2.3}$$

the g_i and f_j being arbitrary functions of x, is a poly-y family.

The scope of any matching algorithm is maximized by choosing the set $\mathcal{G}$ to be the largest group of transformations acting on $\mathcal{F}$, known as its *structure invariance group* and denoted $\mathcal{G}_{\mathcal{F}}$ [24]. For poly-y families this group must not increase the degree of the dependent variable y or its derivatives, so that in general for point transformations it must be contained in the group

$$(x, y) \mapsto \left(\chi(x), \frac{\psi_1(x)y + \psi_2(x)}{\psi_3(x)y + \psi_4(x)}\right), \tag{2.4}$$

where invertibility implies $\chi' \neq 0$ and $\psi_1\psi_4 - \psi_2\psi_3 \neq 0$. Abel equations, for example, admit Eq.(2.4) as their structural invariance group.

With the problem fully specified, we simplify it by projecting both target and input equations onto a canonical form. A subfamily $\mathcal{F}_{can} \subset \mathcal{F}$ together with subgroup $\mathcal{G}_{can} \subset \mathcal{G}$ which preserves $\mathcal{F}_{can}$ is considered a canonical form when there is a subgroup $\mathcal{G}_\pi \subset \mathcal{G}$ (with $\mathcal{G} = \mathcal{G}_\pi \, \mathcal{G}_{can}$) such that for each $\Omega \in \mathcal{F}$ there is a unique $\tau \in \mathcal{G}_\pi$ satisfying $\Omega \circ \tau \in \mathcal{F}_{can}$. Further, a canonical form is particularly useful when we know an efficient algorithm that computes such a τ.

The Abel equation Eq.(2.3) with transformation group Eq.(2.4) can be reduced, using for $\mathcal{G}_\pi$ the transformation Eq.(2.4) with $\chi(x) = x$, to the canonical form,[5]

$$y' = c_3(x)y^3 + c_1(x)y + 1. \tag{2.5}$$

$\mathcal{G}_{can}$ is the structural invariance group of $\mathcal{F}_{can}$, that is

$$(x, y) \mapsto (\chi(x), \chi'(x)\ y). \tag{2.6}$$

The algorithm that determines τ is given in for example [24]. This projection removes the four variables ψ_i from the system, while reducing the number of coefficients to be matched from six to two.

---

[4]Throughout this work, we will employ three notations for equality: ':=' denotes the assignment of a variable as the result of a computation; '≡' denotes the mathematical definition of an unknown quantity in terms of others; '=' denotes an assertion of mathematical equality between quantities which may or may not have been calculated.

[5]Only the Abel Bernoulli class, consisting of equations equivalent via Eq.(2.4) to $y' = y^3$ has no representatives in the form Eq.(2.5).

When the resulting system is overdetermined we can use differential elimination to reduce the problem further to an algebraic system. Applying the general transformation Eq.(2.6) to a target equation in canonical form Eq.(2.5) and using an appropriate μ gives the form of the equation which must be matched to the canonical form of the input equation:

$$y' = c_3(\chi(x))\,\chi'(x)^3 y^3 + \left(c_1(\chi(x))\chi'(x) - \frac{\chi''(x)}{\chi'(x)}\right) y + 1. \tag{2.7}$$

Equation Eq.(2.2) now splits into a system of two equations for one unknown χ:

$$\tilde{c}_3 = (c_3 \circ \chi)\,{\chi'}^3, \tag{2.8}$$

$$\tilde{c}_1 = (c_1 \circ \chi)\,\chi' - \frac{\chi''}{\chi'}. \tag{2.9}$$

We call χ the *equivalence function*, since it is the only unknown remaining in the equivalence problem after it has been converted to canonical form. Replacing (2.9) by the logarithmic derivative of (2.8) plus three times (2.9), we eliminate χ'':

$$\frac{\tilde{c}_3'}{\tilde{c}_3} + 3\,\tilde{c}_1 = \left(\left(\frac{{c_3}'}{c_3} + 3\,c_1\right) \circ \chi\right) \chi'. \tag{2.10}$$

We then replace (2.8) by the cube of Eq.(2.10) divided by (2.8), obtaining a new equation free of χ':

$$\frac{\left(\tilde{c}_3' + 3\tilde{c}_3\,\tilde{c}_1\right)^3}{\tilde{c}_3^4} = \left(\frac{\left({c_3}' + 3c_3c_1\right)^3}{{c_3}^4}\right) \circ \chi. \tag{2.11}$$

Similarly, dividing the logarithmic derivative of Eq.(2.11) by Eq.(2.10) gives a second equation algebraic in χ which we use to replace Eq.(2.10):

$$\frac{3\tilde{c}_3\tilde{c}_3'' + 9\tilde{c}_3^2\tilde{c}_1' - 3\tilde{c}_3\tilde{c}_3'\tilde{c}_1 - 4\tilde{c}_3'^2}{(3\tilde{c}_3\tilde{c}_1 + \tilde{c}_3')^2} = \left(\frac{3c_3{c_3}'' + 9{c_3}^2{c_1}' - 3c_3{c_3}'c_1 - 4{c_3'}^2}{(3c_3c_1 + {c_3}')^2}\right) \circ \chi. \tag{2.12}$$

Equations (2.11, 2.12) retain the form of (2.8, 2.9) in that they manifest the effect of the transformation $\tau = (\chi, \psi)$ on some expression formed from the ODE. Importantly though, the transformation acts on these particular expressions:

$$I_1 \equiv \frac{(3c_3\,c_1 + c_3')^3}{c_3^4}, \quad I_2 \equiv \frac{3c_3{c_3}'' + 9{c_3}^2{c_1}' - 3c_3{c_3}'c_1 - 4{c_3'}^2}{(3c_3c_1 + {c_3}')^2}.$$

as a composition with a single unknown function. As we shall see in the next section, such expressions are known as *invariants*.

Note that each step of the differential elimination must be invertible, to ensure that the resulting system is equivalent to the original. Consequently, if invariants are rational functions of the coefficients and their derivatives, the coefficients of the canonical form equation can always be expressed as rational functions of the invariants and their derivatives, simply by reversing the steps. For Abel equations these expressions are:

$$c_3 = \frac{{I_1'}^3}{{I_1}^4 {I_2}^3}, \quad c_1 = \frac{4I_1'}{3I_1} + \frac{I_2'}{I_2} + \frac{I_1'}{3I_1 I_2} - \frac{I_1''}{I_1'}. \tag{2.13}$$

This inversion, expressing the coefficients of the canonical form of the equation in terms of the invariants, confirms that matching the invariants I_1 and I_2 is sufficient to match the ODEs. A set of absolute invariants having this property is called *a fundamental set of invariants*. More generally, if $c_i(x)$ are the coefficients of a poly-y equation in canonical form, and $I_i(x)$ is a sequence of fundamental invariants which has been obtained from $\{c_i(x)\}$ via a differential elimination process, then by reversing these steps we can express the coefficients $c_i(x)$ as rational expressions in the invariants $I_i(x)$ and their derivatives:

$$c_i(x) = f_i(I_1(x), \ldots, I_n(x), I_1'(x), \ldots, I_n'(x), \ldots). \tag{2.14}$$

Such a fundamental set also has the characteristic that all possible invariants can be expressed as functions of the invariants in the fundamental set and their derivatives.

The remaining problem is to solve the decomposition problem using algebraic methods. When the invariants and equivalence function are rational we can apply the *standard equivalence method*, as described by Liouville in [16]. We start with the system equating the corresponding invariants of the transformed target and input equations.

$$\tilde{I}_1(x) = I_1(\chi), \quad \tilde{I}_2(x) = I_2(\chi), \quad \ldots. \tag{2.15}$$

We then subtract the right hand side from the left hand side and take numerators, obtaining a system of polynomials. When there are no parameters to solve for we can take a greatest common divisor of the resulting polynomials, and solve the result for $\chi(x)$. If there are parameters that must also be determined, we can take resultants between various pairs of polynomials to remove these extra solving variables one at a time. For this approach to be feasible, we must start with an overdetermined system, that is, one having at least one more invariant than unknown variable. As shown in [8], extra tricks can simplify the problem:

- At first, giving a (nonsingular) value to the independent variable x to simplify the search for the parameter values defining the class. Repeating the steps using the newly found parameter values but without substituting a value for x to find the equivalence function $\chi(x)$.

- Giving (nonsingular) values to any symbolic parameters in the input invariants, and then using rational function interpolation to determine the dependency of the target parameters and equivalence function on these input parameters.

This method has proved successful as shown in [8] for matching Abel ODE classes depending on one parameter, but for classes depending on more than one parameter the intermediate expressions created become too large to be feasible.

In the next section, we will describe some properties of invariants that enable a novel method of solving this decomposition.

2.1.1 The structure of invariants

As mentioned above, the standard method for solving the remaining decomposition problem can be too expensive when the class being matched depends on many parameters. In this subsection we demonstrate that groups of invariants typically contain many common factors, a property which suggests a more efficient matching technique.

Let $\mathcal{G}$ be a group of transformations acting on a poly-y family of ODEs $\mathcal{F}$, and $\mathcal{J}(\Omega; x, y)$ a function of an equation Ω in $\mathcal{F}$, its partial derivatives $\Omega_x, \Omega_y, \Omega_{y'}, \ldots$, and $x, y, y', \ldots$. Extending the transformation Eq.(2.1) to the jet space,

$$\tau \equiv (x, y, y', \ldots) \mapsto \left(\chi(x, y), \psi(x, y), \frac{D\psi}{D\chi}, \ldots\right) \tag{2.16}$$

we call $\mathcal{J}$ an *invariant* of $\mathcal{F}$ with respect to $\mathcal{G}$, if there is a function $\varphi_{\mathcal{J}}(\tau)$ such that

$$\mathcal{J}(\mu\,\Omega \circ \tau; x, y) = \varphi_{\mathcal{J}}(\tau)\;\; \mathcal{J}(\Omega; x, y) \circ \tau \tag{2.17}$$

for all equations Ω in the family $\mathcal{F}$, transformations τ in the group $\mathcal{G}$, and integrating factors μ.

The invariant $\mathcal{J}$ is *absolute* when $\varphi_{\mathcal{J}}$ is identically 1, *relative* otherwise. A sequence of relative invariants $\{\mathcal{J}_n\}$ are said to have *weights* n when each corresponding $\varphi_{\mathcal{J}_n}$ can be written as the n^{th} power of a single function ϕ, i.e., $\varphi_{\mathcal{J}_n} = \phi^n$. The product of two relative invariants of weights n and m is a relative invariant of weight $n + m$. Absolute invariants can therefore be constructed as products of powers of relative invariants such that the total weight is zero [6, 11, 20].

As an example, the functions ${c_3}^2$ and ${c_3}^2(c_3' + 3c_3c_1)$ are the first two relative invariants s_3 and s_5 for Abel's DE of the sequence introduced by Liouville in [16], with $\varphi_{s_n}((x, y) \mapsto (\chi(x), \chi'(x)y)) = {\chi'}^{2n}$. Consequently, the quotient ${s_5}^3/{s_3}^5$ is an absolute invariant, the same one we derived in Eq.(2.11).

More generally, if $\mathcal{J}$ is an absolute invariant for $(\mathcal{F}, \mathcal{G})$, where $\mathcal{G} = \{(x, y) \mapsto (\chi(x), \psi(x, y))\}$, and $\mathcal{J}(\Omega)$ is a function of x for all $\Omega \in \mathcal{F}$, then $\frac{d}{dx} \circ \mathcal{J}$ is a relative invariant with

$\varphi_{\mathcal{J}}((x, y) \mapsto (\chi(x), \psi(x, y))) = \chi'(x)$. Therefore, the ratio of the derivatives of two absolute invariants yields a third absolute invariant:

$$\frac{dI_n}{dI_m} = \frac{dI_n(x)/dx}{dI_m(x)/dx}. \tag{2.18}$$

If I_m and I_m are rational in x, this new expression will be as well. Furthermore, any polynomial which occurs to a multiplicity greater than 1 in I_m will also occur in its derivative, so in general, *the new invariant* $\frac{dI_n}{dI_m}$ *will contain as factors many of the same polynomials as the original invariants.* As a corollary, if $\{r_n\}$ are relative invariants of weight n with $\phi = \chi'$, then

$$n r_n {r_m}' - m r_m {r_n}' = r_m r_n \frac{d}{dx} \ln \left(\frac{{r_m}^n}{{r_n}^m} \right)$$

is a relative invariant of weight $n+m+1$. This provides a method of generating new relative invariants from an existing set. If r_n and r_m are polynomial then so will be the new relative invariant. Sequences of polynomial relative invariants derived in this way can be combined to give absolute invariants sharing many of them as common factors.

Given an overdetermined system of rational functions, many of which contain the same polynomial factors but to different degrees, it is possible, through a squarefree factorization, to break these rational functions down into their constituent polynomials. This is the basis for the algorithm which we will describe in the following section.

2.2 Invariant component polynomials

We have seen that the sequence of invariants of a rational ODE are not random rational functions, but are composed as specific products of powers of a common set of polynomials, which we shall call the *invariant component polynomials.* As we shall see, these ICPs can be computed very efficiently; are invariant under equivalence transformations; and because they are fundamentally simpler objects, can be matched far more quickly using a GCD approach than can the invariants themselves. Furthermore, we will prove that the 2-d array of powers encapsulating the relationship between these ICPs and the invariants they comprise is a numerical invariant of the class; we call it the *invariant signature.* The signature is an easy-to-compute and powerful preliminary filter: if two invariant sequences have different signatures, they cannot be equivalent.

A simple example of the type of problem under consideration is to determine the equivalence relation $x \mapsto x^2/(x^2-1)$ between the following pair of invariant sequences:[6]

$$R := \left[\frac{729(9x-1)^3}{50}, \frac{54x(1-2x)}{(9x-1)^2}\right], \quad \tilde{R} := \left[\frac{729(8x^2+1)^3}{50(x^2-1)^3}, -\frac{54x^2(x^2+1)}{(8x^2+1)^2}\right].$$

The polynomial $9x-1$ appears as a factor in both invariants of the first sequence with corresponding powers 3 and -2. With the second sequence, the same pattern 3 and -2 occurs for the polynomial $8x^2+1$ which is thus seen as analogous to $9x-1$. Similarly the polynomial $x(1-2x)$ with powers 0 and 1 in the first sequence has analogue $x^2(x^2+1)$ in the second sequence. The idea at its most basic level is to use these pairs of analogous polynomials instead of the more complicated and higher degree rational functions to search for an appropriate equivalence transformation. While there are complications with this idea, there ends up being a significant net benefit in efficiency to the algorithm as a whole. Before continuing, note one subtlety in the example above: the polynomial x^2-1 with powers -3, 0 appearing in $\tilde{R}$ seems not to have an analogue in the first sequence; but the analogue is there — it is actually the constant polynomial! In order to properly treat such special cases we use homogeneous polynomials in the theory presented below.

2.2.1 Definitions and properties

In the following let $R = [R_i(x)]_{i=1\ldots s}$ be a list of rational functions with coefficients in a field $\mathbf{K}$.

Definition 2.2.1. Let $\{P_j\} \subset \mathbf{K}[x_1, x_2]$ be the set of monic[7] irreducible factors of the expressions $[R_i(x_1/x_2)]_{i=1\ldots s}$, and $m_{j,i}$ be the number of times the factor P_j divides the numerator of $R_i(x_1/x_2)$ minus the number of times it divides the denominator. Define the *R-multiplicity* M_j *of* P_j by

$$M_j \equiv \left[\frac{m_{j,1}}{c_j}, \ldots, \frac{m_{j,s}}{c_j}\right], \text{ where } c_j \equiv \mathrm{GCD}(m_{j,1}, \ldots, m_{j,s}).$$

Call $S_R = \{M_j\}$ the *signature of* R, and define the *invariant component polynomials (ICPs) of* R to be

$$R_M \equiv \prod_{M_j = M} {P_j}^{c_j}, M \in S_R. \tag{2.19}$$

We also say that a polynomial in $\mathbf{K}[x_1, x_2]$ such as R_M has *R-multiplicity* M when all of its monic irreducible factors do.

[6] These are the Liouville invariants (see Chapter 3) of a pair of Abel equations. The first is $y' = (50y^3 - 3(9x-1)y + 1)/(18x(1-2x))$, the second obtained from the first using $x \mapsto x^2/(x^2-1)$.

[7] To define what is meant by monic, the choice of ordering must only be consistent. We therefore assume throughout this article a lexicographical ordering on the terms.

For the invariant sequence R given above, we find $S_R = \{[3,-2],[-1,0],[0,1]\}$, with

$$R_{[3,-2]} = x_1 - \frac{1}{9}x_2, \quad R_{[-1,0]} = {x_2}^3, \quad R_{[0,1]} = {x_1}^2 - \frac{1}{2}x_1x_2.$$

Notice that we can write

$$R_1\left(\frac{x_1}{x_2}\right) = \frac{3^{12}}{50}{R_{[3,-2]}}^3{R_{[-1,0]}}^{-1}{R_{[0,1]}}^0, \quad R_2\left(\frac{x_1}{x_2}\right) = -\frac{4}{3}{R_{[3,-2]}}^{-2}{R_{[-1,0]}}^0{R_{[0,1]}}^1.$$

Proposition 2.2.2. *Each function R_i can be written as a product of the ICPs of R, up to a constant α_i:*

$$R_i\left(\frac{x_1}{x_2}\right) = \alpha_i \prod_{M\in S_R} {R_M}^{m_i}, i = 1\ldots s \tag{2.20}$$

where we have denoted $M = [m_1, m_2, \ldots]$. Moreover, this is the only way to write each of the functions $R_i(x_1/x_2)$ as a product of monic, pairwise relatively prime factors having distinct R-multiplicities.

Proof. Suppose the alternative decomposition:

$$R_i\left(\frac{x_1}{x_2}\right) = \beta_i \prod_{N\in T} {U_N}^{n_i}, i = 1\ldots s.$$

From this equation, any irreducible factor P_j dividing U_N k_j times must divide $R_i(x_1/x_2)$ a total of k_jn_i times; from Eqs. (2.19, 2.20) it divides the same expression c_jm_i times for some $M \in S_R$. Thus $k_jn_i = c_jm_i$ for all i; it follows that $N = M$ and $k_j = c_j$. This being true for all such P_j, we must have $U_N = R_M$.

This proposition shows that the signature and ICPs of R are characterized by Eq.(2.20). We will see next that they are also invariant under rational transformations.

Definition 2.2.3. Let G, H be relatively prime polynomials in $\mathbf{K}[x]$. For homogeneous $P \in \mathbf{K}[x_1, x_2]$ of degree n, denote

$$\overline{P}(x_1, x_2) \equiv \alpha_P P\left(G\left(\frac{x_1}{x_2}\right), H\left(\frac{x_1}{x_2}\right)\right){x_2}^{dn}, \tag{2.21}$$

where $d = deg(P)$, $n = \max(deg(G), deg(H))$, and the constant α_P has been chosen to make $\overline{P}$ monic.

Lemma 2.2.4. *(i) If $Q_1, Q_2 \in \mathbf{K}[x_1, x_2]$ then $\overline{Q_1Q_2} = \overline{Q_1}\ \overline{Q_2}$*
(ii) If $Q_1, Q_2 \in \mathbf{K}[x_1, x_2]$ are relatively prime then so are $\overline{Q_1}, \overline{Q_2}$.

Proof. The first assertion is obvious; the second was shown by Zippel in [26].

Proposition 2.2.5. *Consider the transformed system $\tilde{R} = [R_i(G(x)/H(x))]_{i=1\ldots n}$. Then $S_{\tilde{R}} = S_R$, and for each $M \in S_R$, $\tilde{R}_M = \overline{R_M}$.*

Proof. Let $N_i(x_1,x_2)$,$D_i(x_1,x_2)$ be the numerator and denominator respectively of $R_i(\frac{x_1}{x_2})$. Note that N_i and D_i have the same degree, say d. Then

$$\tilde{R}_i\left(\frac{x_1}{x_2}\right) = \frac{N_i(G(x_1/x_2),H(x_1/x_2)){x_2}^{dn}}{D_i(G(x_1/x_2),H(x_1/x_2)){x_2}^{dn}} = \frac{\alpha_{D_i}}{\alpha_{N_i}}\frac{\overline{N_i}}{\overline{D_i}} = \frac{\alpha_{D_i}}{\alpha_{N_i}}\alpha_i \prod_{M\in S_R} \overline{R_M}^{m_i}.$$

The result now follows from Proposition 2.2.2 and 2.2.4(ii).

For the above example with $G(x) = x^2 - 1$, $H(x) = x^2$, and $M = [3,-2]$, we compute

$$\begin{aligned}\overline{R_{[3,-2]}} &= \alpha R_{[3,-2]}\left(\frac{{x_1}^2}{{x_2}^2}, \frac{{x_1}^2 - {x_2}^2}{{x_2}^2}\right){x_2}^2 \\ &= \alpha\left({x_1}^2 - \left(\frac{{x_1}^2 - {x_2}^2}{9}\right)\right) \\ &= {x_1}^2 + \frac{{x_2}^2}{8} \\ &= \tilde{R}_{[3,-2]}.\end{aligned}$$

2.2.2 The invariant signature as a filter

Using the signature to classify a sequence of invariants is particularly advantageous when they are the invariants of a differential equation. In such a case, there may be a large, possibly infinite, set of prospective target classes (ie. those with known solutions), but if all classes are related as specific instances of a single parameterized class, their signatures will be similar, and in fact the number of such signatures will be finite. For example, the set of possible signatures corresponding to all Abel ODEs with known solutions is quite small, as will be shown in Chapter 3.

Having restricted the set of possible matches with the use of the signature, we can then apply a more precise matching algorithm that only needs to cover this more restrictive set. In other words, we can incorporate the knowledge of the signature into the matching algorithm for specific classes. Some signatures correspond to only a small set of classes, so that this filter restricts the search space dramatically, and makes the corresponding matching algorithm easier to design. For more common signatures, we will find, in general, that the same matching algorithm works for many classes. Finally, if the invariant signature of a given equation does not match that of a known-solvable class, we can discard the equation as unsolvable immediately, without attempting a computationally intensive match to known equations.

Example 2.2.6. In Chapter 3, we present a list of three invariants $L = [L_1, L_2, L_3]$ for Abel equations. The most general Abel inverse Riccati class, represented by the parameterized equation

$$y' = -\frac{dy^3 + ay^2 + by + c}{y - x^2}. \tag{2.22}$$

has the corresponding signature $S_L = \{[-5,0,1],[0,1,-1],[3,-2,-1],[0,0,1],[0,1,0]\}$. The corresponding ICPs for this class, (evaluated at $(x,1)$), are[8]

$$\begin{aligned}
L_{[0,1,0]} &\simeq (dx^6 + ax^4 + bx^2 + c)^2, \\
L_{[-5,0,1]} &\simeq \left(\frac{dab}{3} - \frac{2}{3d} - d^2c - \frac{2}{27}a^3\right)x^6 + \cdots + dc^2 + \frac{2}{3}c - \frac{1}{3}acb + \frac{2}{27}b^3, \\
L_{[3,-2,-1]} &\simeq \frac{1}{27}(a^2 - 3db)(2a^3 - 9dab + 12d + 27d^2c)x^{10} \\
&\quad + \cdots + \frac{1}{27}(b^2 - 3ac)(9acb - 27dc^2 - 12c - 2b^3), \\
L_{[0,1,-1]} &\simeq \left(-\frac{4}{9}a^4 + \frac{64}{27}da^2b - \frac{20}{9}d^2b^2 - \frac{16}{9}da - \frac{8}{3}d^2ca\right)x^8 \\
&\quad + \cdots + \frac{64}{27}acb^2 - \frac{4}{9}b^4 - \frac{20}{9}a^2c^2 - \frac{16}{9}cb - \frac{8}{3}bdc^2, \\
L_{[0,0,1]} &\simeq \left((54c^3)d^5 + \cdots + \frac{40}{27}a^6\right)x^{12} + \cdots \\
&\quad + \left(\left(54c^5\right)d^3 + \cdots + \left(\left(\frac{8}{3}bc^3\right)a^4 + \cdots + \left(\frac{64}{27}cb^3 - \frac{160}{9}c^2 + \frac{40}{27}b^6\right)\right)\right).
\end{aligned} \tag{2.23}$$

By assigning values to the parameters a, b, c, and d we obtain specific classes which in general have the same signature S_L. For most values of the parameters, the same algorithm can be used to match the class and determine the equivalence transformation (see Chapter 3). However, there are some special values for the parameters which yield classes with different signature. For example the class defined by $a = b = 5/4$, $c = d = -1/4$ has signature $\{[-1,0,0],[0,0,1],[0,1,0],[3,-2,-1]\}$. From the infinite set of AIL and AIR classes, only three 0-parameter subclasses have this signature: the one described above, as well as $AIL[b]$, $b = 2$, and $AIR[-2/3]$. Hence we must only check whether an input equation having this signature matches one of these three 0-parameter classes to find its class or determine that it is neither AIL nor AIR. Finally, we can compute the signature of the equation $y' = y^3 + x$ to be $\{[-1,0,0],[1,0,0]\}$, which does not appear among the signatures of any AIL or AIR class, which have been listed in Chapter 3. We can therefore discard this equation as neither AIL nor AIR after essentially only computing its invariants and ICPs.

2.2.3 Constructing new invariants from the ICPs

The ICPs of Definition 2.2.1 are not invariants in the sense of Eq.(2.17), but we can use them to generate new invariants in a surprising way.

[8] The symbol $\simeq$ denotes 'equality up to a constant multiple', as the ICPs are defined only up to a constant. We have omitted some of the terms in these polynomials, in the interest of brevity. See Chapter 3 for a derivation of the explicit values of these ICPs.

Corollary 2.2.7. *Any ICP R_M of degree d transforms under $x \mapsto F(x) = G(x)/H(x)$ according to*

$$\tilde{R}_M(x,1) \simeq R_M(F(x),1)H(x)^d.$$

Proof. Consider equation Eq.(2.21) with $P = R_M$ and $(x_1, x_2) = (x, 1)$, and apply Proposition 2.2.5. The fact that $R_M(G,H) = R_M(G/H,1)H^d$ is a result of the homogeneity of R_M.

Proposition 2.2.8. *If R_M and R_N are ICPs of the same degree, then $J = \frac{d}{dx}\ln\frac{R_M(x,1)}{R_N(x,1)}$ is a relative invariant with $\varphi = F'(x)$ for the transformation $x \mapsto F(x)$.*

Proof. Let $\tilde{R}$ be a list of invariants transformed via any rational equivalence transformation $x = F(x)$ as in Proposition 2.2.5. Then

$$\frac{\tilde{R}_M(x,1)}{\tilde{R}_N(x,1)} \simeq \frac{R_M(F(x),1)}{R_N(F(x),1)}. \tag{2.24}$$

It follows that $\tilde{J}(x) = F'(x)J(F(x))$.

Having constructed new relative invariants, it is natural to combine them with existing ones to generate new absolute invariants, and thus new ICPs. This process can be repeated, leading to successively simpler ICPs, and eventually to the equivalence function itself.

Example 2.2.9. We can create additional invariants from the ICPs for the Abel inverse-Riccati equation Eq.(2.22). Taking for instance $a = b = d = 1, c = 2$, we calculate the ICPs for the corresponding set of Liouville invariants $L = [L_1, L_2, L_3]$. These include:

$$\begin{aligned}
L_{[0,1,0]}(x,1) &= (x^6 + x^4 + x^2 + 2)^2, \\
L_{[-5,0,1]}(x,1) &= x^6 - \frac{12}{65}x^5 + \frac{9}{13}x^4 - \frac{118}{65}x^3 - \frac{27}{65}x^2 - \frac{6}{13}x - \frac{128}{65}, \\
L_{[3,-2,-1]}(x,1) &= x^{10} - \frac{32}{59}x^9 + \frac{593}{118}x^8 - \frac{380}{59}x^7 + \frac{312}{59}x^6 - \frac{1673}{59}x^5 - \frac{117}{59}x^4 - \frac{926}{59}x^3 \\
&\quad - \frac{19}{2}x^2 - \frac{149}{59}x - \frac{290}{59}.
\end{aligned}$$

The ratio of two weight-1 relative invariants, the first derived from $L_{[0,1,0]}$ and $L_{[-5,0,1]}$ as in Proposition 2.2.8, the second being $r_1 \equiv 3c_1 + \frac{d}{dx}\ln c_3$ (*cf.* Eq.(2.10)), is an absolute invariant:

$$A_1 := \frac{1}{r_1}\frac{d}{dx}\ln\left(\frac{L_{[0,1,0]}(x,1)}{L_{[-5,0,1]}(x,1)^2}\right).$$

We now redefine R by adding a modified version of the newly created absolute invariant $A_2 := 1 - \frac{1}{2}A_1$ to the existing list of invariants:

$$\Lambda := [L_1, L_2, L_3, A_2].$$

The calculation determining the ICPs is repeated, and it produces one new ICP:

$$\Lambda_{[0,0,0,1]}(x,1) = \frac{59}{65}\frac{L_{[3,-2,-1]}(x,1)}{L_{[-5,0,1]}(x,1)}A_2 = x^4 + \frac{13}{2}x^2 + \frac{5}{2}.$$

This latest polynomial is useful in that it is only of 4th degree, and is polynomial in x^2. Together with $L_{[0,1,0]}(x,1)$, it can be used to generate a new set of ICPs, all polynomial in x^2. This surprising result allows us to efficiently determine the equivalence function $F(x)$ for a given equation of the same class. We will avoid providing more details on this example here - to do so would spoil the main result in Chapter 3.

2.2.4 Pseudo-invariants and the ICP method

In light of the two propositions above, it makes sense to generalize the concept of an invariant so that the new concept includes ICPs.

Definition 2.2.10. We call a function, or more generally an algorithm, $\mathcal{J}$ a *polynomial pseudo-invariant* of weight w and degree d if there is a degree $d-2w$ homogeneous polynomial P in two variables such that for all equations Ω and transformations $\tau : x \mapsto G(x)/H(x)$ we have $\mathcal{J}(\Omega\circ\tau)(x) = \alpha P(G,H)(G'H - GH')^w$ for some non-zero *constant multiplier* α. More generally, any quotient of two polynomial pseudo-invariants is called a *pseudo-invariant*, its degree and weight being determined multiplicatively.

Using this definition, ICPs are polynomial pseudo-invariants of weight 0, relative invariants of weight w are pseudo-invariants of weight w and degree 0, absolute invariants are pseudo-invariants of weight 0 and degree 0.

The ICP method can be roughly described as a series of steps, starting from the ICPs, and resulting in a sequence of simpler and simpler pseudo-invariants until the set of known pseudo-invariants includes the parameters and the equivalence function. We first develop these steps for the transformed target equation of a given class; having done so, the matching algorithm consists of applying those same steps to an input equation. While this will certainly work if the input equation is exactly the same class as the target representative of the super-class, certain steps may fail for certain values of the parameter values defining the input subclass. This may happen for two reasons: a division which works for a superclass may take the form 0/0 for the subclass, or a GCD may contain extra polynomials for the subclass (equivalently, the result of a numerator or denominator in the input algorithm may divide but not equal the analogous result from the target algorithm). In order to develop a complete algorithm, we must identify these subclasses and include extra branches in the algorithm to account for them. If any of the verification steps fail, the input equation is not equivalent to the target equation.

Definition 2.2.11. Let R be a finite sequence of rational invariants depending on parameters $a_1, \cdots, a_n$. An *ICP equivalence algorithm for* R is an algorithm which takes as input a relative invariant and a finite sequence of polynomials $P_M(x_1, x_2)$ indexed by multiplicities $M \in S_R$, and returns one of two results:

1. A rational equivalence function $\chi(x) = G(x)/H(x)$, where $G, H \in \mathbf{K}[x]$, and a sequence of parameter values $\alpha_1, \cdots, \alpha_n$ for which $P_M = \overline{R_M}_{|\{a_i=\alpha_i\}}$ $\forall\ M \in S_R$. [Recall the definition of $\overline{P}$ in Eq.(2.21).]

2. An assertion that there exist no such χ and $\alpha_1, \cdots, \alpha_n$.

There are a few basic methods of creating new pseudo-invariants:

1. Addition, subtraction of pseudo-invariants with the same weight, degree, and constant multiplier; all of these properties propagate to the result.[9]

2. Multiplication, division; all properties are propagated multiplicatively.

3. Extraction of the root of a perfect power of a polynomial pseudo-invariant; all properties propagate to agree with the above.

4. Computation of the GCD of two polynomial pseudo-invariants of weight 0, by taking the GCD of the associated polynomials P_1 and P_2; the weight remains 0, while the degree follows from the degree of the gcd of P_1 and P_2.[10]

5. Differentiation: generalizing Proposition 2.2.8 the derivative with respect to x of any pseudo-invariant of weight 0 and degree 0 is a pseudo-invariant of degree 0 and weight 1.

Let us now describe some common techniques that are built from these basic steps.

Compute the numerator or denominator of a pseudo-invariant of weight 0. These are polynomial pseudo-invariants (we compute them by computing the numerator and denominator of the associated quotient P_1/P_2, by Lemma 2.2.4).

---

[9]The constant multiplier must be the same so that we still know the formula for the target pseudo-invariant in terms of G and H.

[10]Note that the GCD operation does not apply to pseudo-invariants not of weight 0, since we need Lemma 2.2.4. In fact, every nontrivial linear combination $G + \epsilon H$ can have a common factor with $G'H - GH'$ if we choose G and H appropriately; a simple choice is $G = \epsilon x^2, H = 4x + 4$, the common factor being $x + 2$, and it is clear the same result holds for the trivial combinations G and H themselves.

Determining the factorization of an arbitrary pseudo-invariant is also not an invariant operation. For example, depending on the values of G, H, there may be factors than expected. Or, restricting to a factorization over the associated coefficient field, there may be fewer: $(G-H)(G+H)$ becomes x^2+1 under $G = x$, $H = i$.

Extract sub-ICPs. For example if we have two weight-0 pseudo-invariants which factor in terms of other pairwise relatively prime pseudo-invariants X, Y, Z as $P_1 = X^2Y^6Z$, $P_2 = XY^{-2}$, we can compute X (up to a constant factor) as the product of the polynomial factors of P_1 and P_2 which divide P_1 twice as often as P_2. We denote this $X \simeq [P_1 : P_2]_{[2:1]}$. Similarly, we have $Y^2 \simeq [P_1 : P_2]_{[3:-1]}$ and $Z \simeq [P_1 : P_2]_{[1:0]}$.

Simplify the known relative invariant by dividing out known [or newly discovered] polynomial pseudo-invariants. It only makes sense to keep track of one relative invariant, since any two can be described by choosing one of the pair, as well as their weight-0 quotient. Suppose we have a relative invariant $\mathcal{J}$ for which $\mathcal{J}(\Omega) = r(x)$ where r is rational; then we can write $\mathcal{J}(\tilde{\Omega}) = r(F)F' = (G'H - GH')\frac{n(G,H)}{d(G,H)} \prod_i P_i(G,H)^{e_i}$, where the P_i are known polynomial pseudo-invariants. We can remove the P_i from this expression, obtaining $(G'H - GH')\frac{n(G,H)}{d(G,H)}$. In many cases, n and d are constant, so we obtain $\alpha(G'H - GH')$ for some constant α.

Simplify a weight 1 pseudo-invariant computed as in step 5 above. Given weight-0 pseudo-invariants P_i with degrees d_i, assume the product $P = \prod_i P_i^{e_i}$ has degree 0. Any linear factor having a nonzero multiplicity e_i in P will occur with multiplicity $e_i - 1$ in the derivative $\frac{dP}{dF}$. Better is the logarithmic derivative, in which the corresponding multiplicities are all -1. Finally, we multiply this logarithmic derivative by the bases P_i of all powers in the original product P:

$$\begin{aligned}
D_*(P_1^{e_1} \cdots P_n^{e_n}) &\equiv \left(\prod_j P_j\right) \frac{d}{dx} \ln \prod_i P_i^{e_i} \\
&= \left(\prod_j P_j\right) F' \sum_i e_i \frac{H^{d_i}}{P_i} \frac{d}{dF} \frac{P_i}{H^{d_i}} \\
&= F' \sum_i e_i \, H^{d_i} \left(\frac{d}{dF} \frac{P_i}{H^{d_i}}\right) \prod_{j \neq i} P_j \\
&= F' \sum_i e_i \, H^{d_i} \frac{\frac{dP_i}{dG}}{H^{d_i - 1}} \prod_{j \neq i} P_j \\
&= \frac{G'H - GH'}{H} \sum_i e_i \frac{dP_i}{dG} \prod_{j \neq i} P_j.
\end{aligned}$$

The result is a new polynomial[11] pseudo-invariant of weight 1 and degree equal to the degree d of $\prod_i P_i$. To obtain a pseudo-invariant of weight 0 we then divide by any other pseudo-invariant of weight 1. For example,

$$\frac{D_*(P_1^{e_1} \cdots P_n^{e_n})}{\alpha(G'H - GH')}$$

[11] Due to the degree condition, the leading coefficient in the summation vanishes, so that the sum is divisible by H.

is a polynomial pseudo-invariant of weight 0 and degree $d-2$.

The end game: determine F and the parameters. Suppose we have three polynomial pseudo-invariants, αG and βH ($\alpha, \beta \neq 0$), and P of degree $d > 1$ and weight 0. We use $\frac{\alpha}{\beta}F$ and the F-polynomial $(\beta H)^{-d}P$ in the following:

Lemma 2.2.12. *If F is an unknown function of x, but we know αF for some unknown constant $\alpha \neq 0$, and a Laurent polynomial $P(F(x)) = \sum_{i=-n}^{n} p_i F^i$, we can split $P(F)$ into its terms $p_i F^i$.*

Proof. First define the operator

$$D_F = \left(\frac{d}{dx}\ln(\alpha F)\right)^{-1}\frac{d}{dx} = F\frac{d}{dF}.$$

Noting that $D_F^j(P(F)) = \sum_{i=-n}^{n} i^j p_i F^i$, we obtain the matrix equation,

$$\begin{pmatrix} P(F) \\ \vdots \\ D_F^{2n}(P(F)) \end{pmatrix} = \begin{pmatrix} 1 & \cdots & 1 & \cdots & 1 \\ -n & \cdots & 0 & \cdots & n \\ \vdots & & \vdots & & \vdots \\ (-n)^{2n} & \cdots & 0 & \cdots & n^{2n} \end{pmatrix} \begin{pmatrix} p_{-n}F^{-n} \\ \vdots \\ p_n F^n \end{pmatrix}.$$

The integer matrix is *Vandermonde*, and therefore invertible. Multiplying by its inverse on the left gives a formula for the terms $p_i F^i$ as linear combinations of the expressions $D_F^j(P(F))$. □

Note that the same result holds if we replace F by $\tilde{F} = \tilde{G}/\tilde{H}$ where $\tilde{G} \equiv \alpha(k_1 G + k_2 H)$ and $\tilde{H} \equiv \beta(k_3 G + k_4 H)$ are two relatively prime, linear pseudo-invariants, and the k_i are known. As shown below, after computing $\tilde{F}$, we will need to apply a fractional linear transformation to obtain F.

Corollary 2.2.13. *Given relatively prime $P_i = \alpha_i(a_i G + b_i H)$ for $i = 1 \ldots 3$, with constants a_i, b_i known and $\alpha_i \neq 0$ unknown, we can determine $F = G/H$.*

Proof. F can be obtained by applying a fractional linear transformation to:

$$\frac{P_2 \frac{d}{dx}(P_1/P_2)}{P_3 \frac{d}{dx}(P_1/P_3)} = \frac{(a_1 b_2 - a_2 b_1)}{(a_1 b_3 - a_3 b_1)}\frac{P_3}{P_2}.$$

□

Generally the original Laurent polynomial has an unknown constant multiplying factor. We can remove this by dividing the original Laurent polynomial by one of its terms; let us thus assume that this constant is not there. Taking appropriate ratios of powers of the terms, such as $(P_3 F^3)/(P_1 F)^3 = P_3/P_1^3$, gives a number of expressions in terms of the coefficients. When we have enough of these expressions, we can solve for the parameters of the problem. We can then compute a product of terms which is of the form αF^g, where α is now a known function of the parameters and g is the gcd of the degrees of the terms; dividing by it gives F^g. Generally g is 1 but if not we can usually reduce it by using Laurent polynomials in F resulting from other pseudo-invariants; if this turns out to be impossible it is because the original target equation was not minimal.

Parameterized classes

For some of the above steps the results may differ when applied to certain subclasses of a parameterized class as opposed to the superclass:

- Division: because the divisor may be 0;
- Extraction of sub-ICPs, GCD, numerator, denominator: polynomials which are relatively prime for general values of the parameters may develop a common factor when the parameters take on specific values.

Consequently, after each such step, we must determine for which subclasses the step could possibly fail or return the wrong result, and provide an alternative branch for that case.

Verification

Using the types of steps described above, we have described how to compute values for F and the parameters. Until they are verified, these values are only candidates, in part because they may have been calculated without using the full set of ICPs. Verification is performed by comparing the invariants of the input equation with those of the computed candidate, using the values of the equivalence function and parameters. A sufficient condition for a match is that we have agreement for each invariant in a fundamental set, as we have seen in Section 2.1.

An alternative method for verification would be to compare the ICPs of the input equation to ICPs constructed from the target equation using these candidate values. To do this, we would compute the first s invariants of the target equation, substitute the parameter values, compute the resulting ICPs, and then perform the transformation Eq.(2.21) on the result. We would then check that each constructed polynomial is a constant multiple of the corresponding input ICP. Finally, the resulting constant ratios α_M would need to satisfy

$$\prod_{M \in S_L} {\alpha_M}^{m_i} = 1 \text{ for each } i = 1 \ldots s,$$

where $M = [m_1, m_2, \ldots]$ and s is the number of fundamental invariants. A tweak on this method would be to have all the target ICPs pre-computed, get them from a lookup-table during verification, substitute the candidate parameter values, and refine these ICP values in the case of cancelled factors.

2.3 Minimal equations and invariant sequences

ICP algorithms as described above are capable of resolving the equivalence between ODEs as long as both target and input equations have rational coefficients, and the transformation Eq.(2.6) has a rational equivalence function $\chi(x)$. As for the first condition, note that large families of differential equations which can be solved by equivalence methods come from a class of equations characterized by a member with rational coefficients, since equations with rational coefficients are generally easier to solve. For example, those classes of Abel ODEs known in the literature to be solvable were shown in [9] to have this property.

In order to satisfy the second condition, we use the freedom to transform the independent variable $x \mapsto \chi(\tilde{x})$ to find an equivalent rational sequence which is as simple as possible.

Definition 2.3.1. We call two sequences of rational functions $\{I_i(x)\}$, $\{\tilde{I}_i(x)\}$ equivalent if one can be obtained from the other via an invertible transformation of the independent variable:

$$I_i = \tilde{I}_i \circ \chi.$$

We call a sequence of rational functions $I_i(x)$ *minimal* if among all equivalent rational sequences $I_i(\chi(x))$ the degrees of the functions $I_i(x)$ are minimal.

The following algorithm [19] converts any rational sequence to an equivalent one which is minimal:

Algorithm 2.3.1. (Netto)
Input: A sequence of n rational invariants $\{I_i(x)\}, i = 1 \dots n$.
Output: A rational equivalence transformation $\chi(x)$ and corresponding minimal sequence $\{\tilde{I}_i(\chi)\}$ which satisfies $I_i = \tilde{I}_i \circ \chi$ for all i.

- Define the corresponding numerators N_i and denominators D_i of each invariant I_i.
- For i from 1 to n, define $R_i(x,t) := N_i(x)D_i(t) - N_i(t)D_i(x)$.
- Compute the GCD $R(x,t)$ of the n polynomials $R_i(x,t)$.
- Choose any two coefficients of powers of t in $R(x,t)$, say $R_1(x), R_2(x)$, whose ratio $R_1(x)/R_2(x)$ is non-constant. This ratio is a value for χ that minimizes the degrees of $\tilde{I}_i$ in $I_i = \tilde{I}_i \circ \chi$.

- For each i, determine $\tilde{I}_i(\chi)$ by eliminating x from $I_i(x)$ using $\chi - R_1(x)/R_2(x) = 0$.

The result is minimal and unique up to a fractional linear transformation due to Lüroth's Theorem. The non-uniqueness is reflected in the choice of R_1, R_2; more generally we could choose for these functions any relatively prime pair of linear combinations of the t-coefficients of $R(x,t)$.

A given invariant sequence is minimal if and only if the polynomial $R(x,t)$ computed during Algorithm 2.3.1 contains only the linear factor $x-t$, or equivalently, when the system

$$I_1(x) = I_1(t),\ \ldots,\ I_n(x) = I_n(t) \tag{2.25}$$

admits only the one solution $x = t$. We may also consider a representative of a super-class which is minimal when its parameters are general but not when they take specific values. In this case minimal representatives for all the subclasses can be determined by finding those parameter values which cause the quotients $Q_i(x,t) = R_i(x,t)/R(x,t)$ to acquire common factors.

Example 2.3.2. Consider the sequence of rational functions:

$$L := \left[\frac{729}{50}\frac{(9bx^2+9-x)^3}{x^3}, 54\frac{(x^2+b)(x-2x^2-2b)}{(x-9x^2-9b)^2}\right].$$

From this we compute $R(x,t) = x - t$, which shows that for arbitrary b, the sequence L is minimal. However, the x-resultant between the quotients $Q_i(x,t)$ contains the following factors not depending on t: $(b-1)(b+1)$. This shows that L is not minimal precisely when $b = \pm 1$. When $b = 1$, we have

$$L := \left[\frac{729}{50}\frac{(9x^2+9-x)^3}{x^3}, 54\frac{(x^2+1)(x-2x^2-2)}{(x-9x^2-9)^2}\right].$$

In this case, $R = (x^2+1)t - x(t^2+1)$. We choose $R_1 = x^2+1, R_2 = -x$, to give $\chi = -x - x^{-1}$ and $L = [-\frac{729}{50}(9\chi+1)^3, -54\chi(2\chi+1)/(9\chi+1)^2]$.

We now show that using a target equation with minimal invariants ensures that all rational coefficient members of the class can be matched via a rational equivalence function.

Definition 2.3.3. We call a member of a poly-y family of equations *rational* when it has coefficients rational in x, and an equivalence class of equations *rational* when at least one of its member equations is rational.

Definition 2.3.4. We call a rational poly-y equation Ω *minimal* when its fundamental invariant sequence $I_1(x), \ldots, I_n(x)$ is minimal.

Let $\mathcal{F}$ be a family of poly-y equations, divided into equivalence classes under the action of a transformation group $\mathcal{G}$ which is a subset of those of the form Eq.(2.4). Let $\{\mathcal{J}_i\}_{i=1\ldots n}$ be a fundamental set of invariants with respect to the action of $\mathcal{G}$ on $\mathcal{F}$, with values which are functions of x, and assume $\mathcal{F}$ admits a fundamental canonical form $\mathcal{F}_{can}$ with corresponding transformation group $\mathcal{G}_{can}$ which admits arbitrary transformations of the independent variable $x \mapsto \chi(x)$.

Theorem 2.3.5. *Every non-constant-invariant rational class $\mathcal{C}$ has a minimal representative. Suppose $\Omega(x,y) \in \mathcal{C}$ is rational with invariant values $\{I_i = \mathcal{J}_i(\Omega)\}_{i=1\ldots n}$. The following conditions are equivalent:*
(i) Ω is a minimal representative for $\mathcal{C}$;
(ii) the only equivalence transformation which leaves Ω unchanged is the identity;
(iii) all rational members of $\mathcal{C}$ can be generated from Ω using rational forms of χ in Eq.(2.4).

Proof. *Existence of a minimal equation $\tilde{\Omega}$:* We apply Algorithm 2.3.1 to obtain a minimal sequence of invariants for the class, and then use Eq.(2.14) to obtain the coefficients of the corresponding minimal representative.

(i) $\Leftrightarrow$ (ii): Let τ_1 be the unique equivalence transformation sending Ω to its fundamental canonical form $\overline{\Omega}$. Let τ_2 be a transformation in $\mathcal{G}_{can}$ which sends x to $\chi(x)$. Then $x = \chi(t)$ is a nontrivial solution of Eq.(2.25) if and only if ${\tau_1}^{-1} \circ \tau_2 \circ \tau_1$ is a nontrivial equivalence transformation preserving Ω.

(i) $\Rightarrow$ (iii): Let $\tilde{\Omega}$ be in the same class as Ω. Then $\tilde{\Omega}(x,y) = \mu\,\Omega(\chi(x), \psi(x,y))$ for some χ, ψ, μ, and $\mathcal{J}_i(\tilde{\Omega}) = \mathcal{J}_i(\Omega) \circ \chi$ for all i. If Ω is also rational, then by Algorithm 1 and the accompanying uniqueness result, there is a rational function $\tilde{\chi}$ satisfying $\mathcal{J}_i(\tilde{\Omega}) = \mathcal{J}_i(\Omega) \circ \tilde{\chi}$ for all i, since Ω is minimal. Then $I_i \circ \tilde{\chi} \circ \chi^{-1} = I_i$ for all i. Minimality of Ω means that $\tilde{\chi} \circ \chi^{-1}$ is the identity, and hence χ is rational.

(iii) $\Rightarrow$ (i): Use Algorithm 1 to find a rational χ and minimal rational sequence $\tilde{I}_i(x)$ satisfying $I_i = \tilde{I}_i \circ \chi$. Use Eq.(2.14) to obtain $\tilde{\Omega}$ such that $\tilde{I}_i = \mathcal{J}_i(\tilde{\Omega})$ for all i. By (iii), there exists a rational $\widetilde{\chi}$ such that $\tilde{\Omega}(x,y) = \mu\,\Omega(\widetilde{\chi}(x), \psi(x,y))$ for some ψ, μ and so $\mathcal{J}_i(\tilde{\Omega}) = \mathcal{J}_i(\Omega) \circ \widetilde{\chi}$, thus $I_i \circ \chi^{-1} = I_i \circ \widetilde{\chi}$ for all i. So χ has degree 1, and $\{I_i\}$ is minimal.

The use of minimal representatives as target equations for the matching algorithms has three advantages:

- An ICP algorithm which finds all rational equivalence transformations matches all rational coefficient ODEs in the class.
- It is possible to find an ICP algorithm which finds the equivalence function. In contrast, when the target equation is not minimal, it is only possible to find $\chi(x)$,

where $\Omega \circ \chi^{-1}$ is minimal.

- The degrees of the invariants are as small as possible, which improves the efficiency of the resulting algorithms.

Furthermore, if we find an ICP algorithm which finds the equivalence function, this is a proof that the target equation is minimal.

The minimality of a representative for a given parameterized class may not be preserved under certain parameter substitutions, so that using this representative as the target equation algorithm will not lead to a complete ICP equivalence algorithm for the class. To prevent this, we must split all parameterized classes into minimal subclasses.

Definition 2.3.6. Given a poly-y family of ODEs $\mathcal{F}$ in the context of an equivalence transformation group $\mathcal{G}$, a subclass $\mathbf{C}_i$ of the super-class $\mathbf{C}$ of $\mathcal{F}$ is *minimal* if it can be represented by a single parameterized poly-y representative ODE Ω_i with possible restrictions on the parameters, such that any allowable parameter evaluation taking $\mathbf{C}_i$ to a subclass preserves the minimality of the representative Ω_i.

Given a minimal parameterized representative for any super-class $\mathbf{C}$, the condition for the minimality of the representative under a parameter substitution is that the quotients $Q_i(x,t)$ are relatively prime. The set of parameter values which satisfy this condition define a subset of $\mathbf{C}$ which form a minimal class. The complementary set of parameter values which violates this condition defines a finite set of subclasses of $\mathbf{C}$ depending on one fewer parameter, for which a new minimal representative must be determined. Repeating this process, we eventually obtain a split of the superclass into a finite disjoint union of minimal subclasses.

The ICP method involves developing a separate ICP equivalence algorithm for each minimal subclass.

2.4 Specializing ICP algorithms to subclasses

The ICPs are not always invariant under the substitution that defines the subclass. That is, applying the parameter substitution defining the subclass into the set of ICPs for the generic class does not necessarily yield the same result as computing the ICPs of the subclass directly, due to the fact that nominally relatively prime ICPs can acquire common factors via this substitution. We say that the structure of the ICPs changes under such a substitution. There are two ways of dealing with this problem. We could generate a new ICP algorithm for each subclass of a superclass for which the ICPs have a different structure. However, this may not be feasible if the subclass in question can not be easily parameterized, which can often happen in practice. Also, this may be duplicating work, both before and during runtime. An alternative approach, which we describe below, is to regenerate the values of the ICPs that are expected for the ICP algorithm of the superclass, as in the definition from the previous section.

2.4.1 Invariant component polynomials for parameterized classes

Suppose we have an ICP algorithm ϵ for a given parameterized invariant sequence, and the ICPs are not invariant under some parameter substitution applied to this sequence, as described in the previous subsection. We will use the notation *parent* and *child* to denote the relation between this superclass and the subclass obtained from it with a given parameter substitution.

One explanation for why the ICP structure can differ between a parent and child class is that the parameter substitution, which we call π, which takes a parent class to its child class does not necessarily commute with the computation, which we call ρ, of the ICPs from the invariants. In the following diagram then, the mappings $\pi \circ \rho$ and $\rho \circ \pi$ can differ. In particular, we distinguish between the child ICPs, those generated from the child invariants; and the *substituted parent ICPs*, those expressions generated from the parent ICPs by the substitution which generates the child class. In order for the ICP algorithm ϵ to match the child class, it requires the latter as input, not the former.

Figure 2.1: Relationship between parent ICPs and child ICPs

$$\begin{tikzcd}
\text{parent invariants} \arrow[r, leftrightarrow, "\rho", "\gamma"'] \arrow[dd, "\pi"'] & \text{parent ICPs} \arrow[r, "\epsilon"] \arrow[d, "\pi"'] & {[F(\tilde{x}), p_1, p_2, \ldots]} \arrow[d, "\pi"'] \\
 & \text{substituted parent ICPs} \arrow[r, "\epsilon"] \arrow[dl, "\gamma"] & \pi([F(\tilde{x}), p_1, p_2, \ldots]) \\
\text{child invariants} \arrow[r, leftrightarrow, "\rho", "\gamma"'] & \text{child ICPs} \arrow[u, dotted, "?"] &
\end{tikzcd}$$

Example 2.4.1. Let us consider the AIR classes defined by applying the substitution $\rho = \{a = b = 5/4, c = d = -1/4\}$ of Example 2.2.6 to the ODE Eq.(2.22), obtaining:

$$y' = \frac{1}{4} \frac{y^3 - 5y^2 - 5y + 1}{y - x^2}. \tag{2.26}$$

Using the notation introduced above, we call Eq.(2.22) the parent class and Eq.(2.26) the child class. Let us denote by P and C their respective Liouville invariant sequences. The

ICPs of order 3 for the child class are:

$$\begin{aligned} C_{[-1,0,0]} \simeq (x^2+2x-1)^6, \quad C_{[0,0,1]} \simeq 31x^4-8x^3+62x^2+8x+31, \\ C_{[0,1,0]} \simeq (x^2+1)^2(x^2-2x-1)^2, \quad C_{[3,-2,-1]} \simeq (x^2-x+2)(2x^2+1x+1). \end{aligned} \tag{2.27}$$

The ICPs for the parent class were listed in Eq.(2.23), and as required by their definition, are pairwise coprime. The substituted parent ICPs however, obtained from Eq.(2.23) through the substitution ρ, are all divisible by the common factor x^2+2x-1:

$$\begin{aligned} P_{[0,1,0]}|^{a=b=5/4}_{c=d=-1/4} &\simeq (x^2+1)^2(x^2+2x-1)^2(x^2-2x-1)^2, \\ P_{[-5,0,1]}|^{a=b=5/4}_{c=d=-1/4} &\simeq (x^2+2x-1)^3, \\ P_{[3,-2,-1]}|^{a=b=5/4}_{c=d=-1/4} &\simeq (x^2-x+2)(2x^2+x+1)(x^2+2x-1)^3, \\ P_{[0,1,-1]}|^{a=b=5/4}_{c=d=-1/4} &\simeq (x^2+2x-1)^4, \\ P_{[0,0,1]}|^{a=b=5/4}_{c=d=-1/4} &\simeq (31x^4-8x^3+62x^2+8x+31)(x^2+2x-1)^4. \end{aligned} \tag{2.28}$$

Because they are obtained simply by substitution from the original ICPs Eq.(2.23), the expressions in Eq.(2.28) would suffice to determine the matching, using the ICP equivalence algorithm for the parent class ϵ. The missing step in the matching process is the determination of these substituted parent ICPs Eq.(2.28) from the child ICPs Eq.(2.27).

2.4.2 Determination of substituted parent ICPs from child ICPs

Having seen the effect of the change in ICP structure between the parent and child classes, and with the aim of finding of solution, we now turn to the cause. In short, this change occurs when parameter evaluation causes hitherto relatively prime ICPs to acquire a common factor.

To see this, take note of the difference between the substituted parent ICPs Eq.(2.28) and the child ICPs Eq.(2.27): those factors common to more than one of the substituted parent ICPs, namely x^2+2x-1, have been multiplied together in the child ICPs, forming a single ICP with a new multiplicity: $C_{[-1,0,0]}$. We call such factors *cancelled factors*, because in some cases they can appear in both the numerator and denominator of one of the parent ICPs, only to disappear when the expressions are normalized. The entire task of reconstructing the substituted parent ICPs is thus to identify such cancelled factors and determine in which of the substituted parent ICPs they occurred, and to which powers.

First note that it is straightforward to go the other way, that is determine the child ICPs from the substituted parent ICPs. In Figure 2.1 we see this as the composition of the function γ which computes the child invariants by gluing together the polynomials according to Eq.(2.20), with the function ρ which computes the ICPs from the invariants. Indeed, for each cancelled factor X, the product of its multiplicity and power within the child ICP is the simply the sum of the associated quantities for each of the substituted parent ICPs in which it appears:

$$2 \cdot [0,1,0] + 3 \cdot [-5,0,1] + 3 \cdot [3,-2,-1] + 4 \cdot [0,1,-1] + 4 \cdot [0,0,1] = 6 \cdot [-1,0,0]. \quad (2.29)$$

So, the important information representing the difference between the child and substituted parent ICPs has two parts: the identity of the cancelled factors, in this case only the one: $x^2 + 2x - 1$; and the corresponding sequence of powers to which each cancelled factor is raised in each of the substituted ICPs: 2,3,3,4,4. Let us formalize this idea in a definition:

Definition 2.4.2. Assuming a given order $P_1, \ldots, P_n$ for the parent ICPs, we define the *type* of a cancelled factor X under substitution π to be the list of powers $[e_1, \ldots, e_n]$, such that X divides $\pi(P_i)$ a total of e_i times. For brevity, we often denote this by $e_1 \cdots e_n$.

Having now introduced the necessary concepts, we will now return to the original problem: how to determine the substituted parent ICPs from the child ICPs. Or, phrased in terms of the ongoing example,

Given the power 6 and multiplicity $[-1,0,0]$ *of the cancelled factor* $x^2 + 2x - 1$ *with respect to the child ICPs, how can we determine that it has type 23344 with respect to the substituted parent ICPs Eq.(2.28)?*

Let us start by noting some simple strategies that can help:

1. Any factor which appears only in one of the substituted parent ICPs will appear with the same multiplicity in the child ICPs. Therefore if the multiplicity of a polynomial in the child ICPs coincides with one of the parent ICPs, it makes sense to start by assuming that it is not a cancelled factor.

2. Any child ICP multiplicity can only result from a finite number of factor types. These types can be calculated in advance and stored in a lookup table, which can then be accessed at run time, whereupon the possible types can be checked in sequence. In many cases a given multiplicity can arise only from one type, so that this step becomes trivial. This would in general require an extra level of iteration for each cancelled factor.

Example 2.4.3. Continuing with Example 2.4.1, there is only one cancelled factor. The lookup table (see Chapter 3) has two entries of interest: The multiplicities $2N \cdot [-1,0,0]$ arise from corresponding types $[2, N, N, 2N-2, 2N-2]$ for $1 \le N \le 4$, and $(2N+3) \cdot [-1,0,0]$ from types $[0, N, N-1, 2N-2, 2N-3]$ for $N = 2$ or 3. Given that the power of the cancelled factor is 6, we must have either $2N$ or $2N+3$ dividing 6. The latter case is impossible for $N = 2$ or 3; the former gives $N = 1$ or 3, ie. types 21100 or 23344, for the cancelled factor $x^2 + 2x - 1$.

There are a few ways in which the situation can be more complicated:

1. There are multiple cancelled factors, having different types but the same multiplicity. We would need to decide how to split them, determining which subfactor is of which type.

2. The cancelled factor disappears altogether from the invariants, instead of appearing as a new ICP. This would happen if a computation such as that in Eq.(2.29) gave as a result the multiplicity $[0, 0, 0]$. For the AIR superclass, this is impossible, as shown in Chapter 3.

3. The multiplicity of a cancelled factor is one of the multiplicities of the parent ICPs.

In either of cases 2 or 3 it would not be obvious that there is even a problem. A comprehensive strategy would need to try the algorithm assuming these problems do not occur, then if it fails, retry assuming they do.

Fortunately, the classes where these more specialized problems do occur are relatively rare, depend on fewer parameters, and can be determined in advance, so that specialized algorithms can be built for them. We will exhibit two examples here. For a more complete treatment we refer the reader to Chapter 3.

Example 2.4.4. (Case 1) Consider the AIR equation

$$y' = \frac{(y-1)(p^2-y)((np+m)y - p(pm+n))}{(p^2-1)^2(y-x^2)}$$

and its associated invariants. When $n = 2$, $x + p$ is a cancelled factor of type 22222. Independently, when $m = -4$, $x + 1$ is a cancelled factor of the same type. However, if we have $p = i = \sqrt{-1}$ as well as $n = 2$, $m = -4$, the type of $x + 1$ changes to 23344. The multiplicity of both factors is $[-1, 0, 0]$, so that the new ICP becomes

$$L_{[-1,0,0]} \simeq (x+1)^6(x+i)^4.$$

In this case, since one of the remaining ICPs is only of degree one,

$$L_{[-5,0,1]} \simeq 41x + 9 - 40i,$$

we could use the prescription mentioned in the previous section to obtain a new absolute invariant of lower degree:

$$A = \frac{1}{r_1}\frac{d}{dx}\ln\left(\frac{L_{[-1,0,0]}}{(L_{[-5,0,1]})^{10}}\right).$$

Instead of developing a distinct matching procedure for this and other such subclasses however, this particular class happens to be easily matched (ie the substituted parent ICPs are found) using an algorithm developed for a more general child class. Indeed, many of these parent ICP determining algorithms are similar, and so it makes more sense to combine them rather than keeping them separate.

Example 2.4.5. (Case 3) Another AIR subclass is represented by the equation

$$y' = \frac{6(y^2+U)}{(1-x^2)y+1+x^2}.$$

The ICPs for this equation have a cancelled factor $x-1$ of type 20022 and multiplicity $[0,1,0]$, which is already the multiplicity of one of the existing ICPs. The child ICP then becomes

$$L_{[0,1,0]} \simeq ((x^4+1)(1+U)+2(1-U)x^2)^2(x+1)^2(x-1)^4,$$

whereas the corresponding substituted parent ICP was

$$P_{[0,1,0]} \simeq ((x^4+1)(1+U)+2(1-U)x^2)^2(x+1)^2(x-1)^2.$$

In this case we use two of the remaining ICPs, which are both of degree 6, to create a new invariant:

$$A := \frac{1}{r_1}\frac{d}{dx}\ln\left(\frac{L_{[0,1,-1]}}{L_{[-5,0,1]}}\right).$$

Defining the new invariant sequence $\Lambda = [L_1, L_2, L_3, A]$, and recomputing gives the cancelled factor $\Lambda_{[0,2,0,1]} \simeq (x-1)^2$ as a new ICP, from which we are able to generate the parent ICP using $P_{[0,1,0]} := L_{[0,1,0]}/\Lambda_{[0,2,0,1]}$.

2.4.3 Summary of strategy for determining the types of cancelled factors

In general, methods of reconstructing the substituted parent ICPs from the child ICPs depend just on the list of types (and degrees) of its cancelled factors. We therefore adopt the following approach as a general method for managing cancelled factors:

1. Determine all possible types of cancelled factors.

2. For each cancelled factor type, calculate the associated multiplicity with respect to the parent class. Determine which of the types have a unique multiplicity, which is also not one belonging to the parent class. Record these types and corresponding unique multiplicities into a lookup table, to be used at runtime.

3. For each non-unique multiplicity, develop a sub-algorithm to determine the existence and value and type of the cancelled factor.

4. Determine further sub-algorithms to tackle special cases, such as:
   (a) multiple cancelled factors
   (b) cancelled factor multiplicity identical to existing parent multiplicity
   (c) cancelled factor not present in child ICPs (multiplicity consists of only 0).

2.5 Summary of the ICP method

Theorem 2.5.1. *Consider the equivalence relation defined by the action of a group of transformations $\mathcal{G}$ on a poly-y family of solvable ODEs $\mathcal{F}$, represented by a parameterized ODE $\Omega_{\mathcal{F}}$. Let $\mathbf{C}$ be the class of all equations in the family. If the following are given, then together the algorithms described therein can be used to solve any input rational-coefficient equation Ω in the family $\mathcal{F}$.*

- *A choice of absolute invariants $\mathcal{J}_1, \ldots, \mathcal{J}_N$, rational functions in the coefficients of the ODE and their derivatives, which contains a fundamental set of invariants.*
- *An algorithm which computes for any input equation Ω the sequence of invariant values $[I_1, \ldots, I_N]$, where $I_i = \mathcal{J}_i(\Omega)$, and records the transformation which brings Ω to its canonical form.*
- *An algorithm which computes the signature and ICPs for a given sequence of rational invariants.*
- *A classification of the class $\mathbf{C}$ into a finite disjoint union of minimal subclasses $\mathbf{C}_1, \ldots, \mathbf{C}_z$, and a corresponding set of minimal representative equations in canonical form, $\{\Omega_1, \ldots, \Omega_z\}$.*
- *For each minimal representative Ω_μ, the signature S_I and ICPs $P_M, M \in S_I$ for the representative sequence $I = [I_1, \ldots, I_N]$.*
- *A table containing all possible types and multiplicities of cancelling factors with respect to the standard ICPs $P_M, M \in S_{\mathcal{I}}$ of some minimal equation Ω_μ. These are organized into two sections: those multiplicities which come from unique types, and those which come from many different types.*
- *An algorithm which replaces all possible cancelled factors whose type is determined uniquely from its multiplicity, producing an 'adjusted' signature and 'adjusted' polynomial pseudo-invariants.*

- *The list of all possible 'adjusted' signatures $\mathcal{S} = \{S\}$ of the subclasses of* $\mathbf{C}$*, from which the multiplicities with unique types have been removed.*

- *Preliminary algorithms for each adjusted signature $S \in \mathcal{S}$, which replace any extra cancelled factors, converting the adjusted polynomial pseudo-invariants to the parent ICPs for the correct minimal representative Ω_i for the class of the input equation. This may involve more than one possibility; each is tried in turn.*

- *For each minimal representative Ω_i, an ICP equivalence algorithm which computes the equivalence function F and parameter values from the invariants and ICPs for the transformed, substituted equation $\tilde{\Omega}_i$.*

- *A classification of the class* $\mathbf{C}$ *into a finite disjoint union of subclasses* $\mathbf{c}_j$*, and corresponding representative equations Ω_j and solutions $y = Z_j(x, k)$.*

Proof. The complete algorithm is as follows:
Given an input ODE Ω,

1. Compute $I = [I_1(\Omega), \ldots, I_N(\Omega)]$.

2. Compute the corresponding signature S_I and ICPs $P_M, M \in S_I$.

3. Check if the multiplicities appearing in the signature are among the multiplicities in the signature of one of the representative classes C_i, or among the possible cancelled multiplicities in the table. If not, return FAIL.

4. Apply the algorithm which replaces the cancelled factors whose multiplicity comes from a unique type.

5. Apply the preliminary algorithm for the resulting adjusted signature, to obtain candidate substituted parent ICPs $\pi(\Omega_i)$ and parent signatures for a number of minimal representatives Ω_μ.

6. For each candidate found above, apply the ICP equivalence algorithm for Ω_μ. If any is successful, we obtain an equivalence function and defining parameters, along with the corresponding minimal representative. If all fail, return FAIL.

7. Determine the solution class $\mathbf{c}_j$, apply the inverse of the linking transformation noted above, followed by the equivalence transformation and the parameter substitutions to the solution $y = Z_j(x, k)$, obtaining the solution to the input equation Ω.

Chapter 3

An Equivalence Algorithm for Solvable Abel ODEs

3.1 The equivalence problem for Abel equations

The ordinary differential equation attributed to Abel following his pioneering work in [1] takes many forms, the most general being known as Abel's equation of the second kind:

$$(g_1 y + g_0)y' = f_3 y^3 + f_2 y^2 + f_1 y + f_0, \tag{3.1}$$

where $y \equiv y(x)$, and g_0, g_1, f_0, f_1, f_2, and f_3 are analytic functions of x. Abel himself considered the special case where $f_3 = 0$; the further specialization $g_1 = 0$ yields a Riccati equation, which itself is equivalent to a linear equation of second order. However, we do not know of any way to linearize the general Abel equation.

In a previous work [9] we described two classes of Abel ODEs labelled Abel inverse linear (AIL) and Abel inverse Riccati (AIR). These classes consist of equations related through an equivalence transformation of the form

$$(x, y) \mapsto \left(\chi(x), \frac{\psi_1(\tilde{x})\tilde{y} + \psi_2(\tilde{x})}{\psi_3(\tilde{x})\tilde{y} + \psi_4(\tilde{x})}\right), \tag{3.2}$$

to either the representative AIL equation:

$$y' = \frac{a_3 y^3 + a_2 y^2 + a_1 y + a_0}{(s_1 x + s_0)y + r_1 x + r_0} \tag{3.3}$$

or the representative AIR equation:

$$y' = \frac{a_3 y^3 + a_2 y^2 + a_1 y + a_0}{(s_2 x^2 + s_1 x + s_0)y + r_2 x^2 + r_1 x + r_0}. \tag{3.4}$$

The meaning of notation inverse linear and inverse Riccati is that by switching the roles of the dependent and independent variables y and x these representative equations are converted to first order linear and Riccati equations respectively.

The AIL and AIR classes of equations are interesting for several reasons. First, we note that together with some other degenerate subcases, these families contain almost all the solvable Abel equations found in the literature, as shown in [9]. Moreover, general formulas for the integration of the AIR class in particular have recently been discovered [7, 22]. Finally we should note that the work [7] presents a previously unknown connection between AIR and a certain type of 2nd order linear equation having hypergeometric solutions.

In Chapter 2, we described a general method for matching parameterized differential equations, based on decomposing the invariants into invariant polynomials. We will now follow the prescription given, obtaining a set of algorithms for solving all rational coefficient member equations of the complete two-parameter AIL and three-parameter AIR superclasses. A practical implementation has been developed for the Maple computer algebra system. These routines represent a significant improvement in efficiency over the previous method described in [8] which was developed for certain 1-parameter AIL and AIR subclasses. Examples will be given.

We start by describing in Section 3.2 a new and more useful set of differential invariants for the general Abel differential equation. In Sections 3.3 and 3.4 we apply the techniques described Chapter 2 to the AIL and AIR classes, developing three building blocks for the subsequent algorithms:

- A set of minimal representative equations for the AIL and AIR classes.
- An explicit derivation of the parameterized values of the invariants for the AIL and AIR classes.
- The associated invariant component polynomials for each class.

Section 3.5 describes in detail the main matching algorithm. For those subclasses which give rise to cancelled factors, we present in Section 3.6 extra sub-algorithms for replacing these cancelled factors. In Section 3.7 we give a brief description of the implementation and demonstrate a few examples.

3.2 Canonical forms, invariants, and Liouville's equivalence method

As described in Chapter 2, classifying an Abel equation involves generating its canonical form, calculating a set of invariants defined for that form, and then comparing the relations among those invariants with those of known classes. A preliminary form for Eq.(3.1) is the Abel equation of the first kind,

$$y' = f_3 y^3 + f_2 y^2 + f_1 y + f_0, \quad f_3 \neq 0, \tag{3.5}$$

obtained using the transformation $y \mapsto 1/y - g_0/g_1$ when $g_1 \neq 0$. Starting from Eq.(3.5), a further simplification is obtained through $y \mapsto y - \frac{1}{3} f_2/f_3$, which makes the coefficient of y^2 vanish. Finally, assuming $f_2 = 0$ (and $f_0 \neq 0$), the transformation $y \mapsto y f_0$ yields the *rational normal form*:[1]

$$y' = c_3(x) y^3 + c_1(x) y + 1, \quad c_3 \neq 0. \tag{3.6}$$

Other canonical forms, notably those suggested by Appell [5],

$$Y'(x) = (Y(x)^3 + J(x))X'(x), \quad X' \neq 0, \tag{3.7}$$

and Liouville [17],[2]

$$z_1{}'(x) + \frac{t'(x)}{T(x)} \left(-\frac{z_1(x)^3}{t(x)^2} + \frac{1 - T(x)}{3t(x)} z_1(x) + \frac{1}{t(x)} \right) = 0, \quad t', T \neq 0, \tag{3.8}$$

can also be obtained from Eq.(3.5) by changing the dependent variable using $y \mapsto \psi_1(x) y + \psi_2(x)$ and renaming (x, y) to (X, Y) and (t, z_1) respectively.

The group of transformations which preserves each form is known as its *structure invariance group*. For the most general Abel equation Eq.(3.1) this group consists of all the equivalence transformations of the form Eq.(3.2); those for the canonical forms are more restricted. Structure invariance transformations for Abel equations of first kind may depend only on χ, ψ_1 and ψ_2, with $\psi_3 = 0$ and ψ_4 constant; those for the rational normal form, $(x, y) \mapsto (\chi(x), \chi'(x) y)$; and those for the canonical forms of Appell and Liouville, $(x, y) \mapsto (\chi(x), y)$, depend only on χ. Assuming that both input and representative equations take one of these last three canonical forms, the equivalence problem for Abel equations reduces to determining this function χ. We shall call χ the *equivalence function* for the transformation Eq.(3.2).

The functions J, X', t, T of equations (3.7, 3.8) are examples of absolute invariants as described in Subsection 2.1.1. Whereas J and X' can be computed from Eq.(3.1) only by taking roots of integrals, t and T can be computed with just derivatives. We will review their derivation and show that they form part of a sequence.

---

[1] This form was suggested in [24]. If $f_2 = f_0 = 0$, Eq.(3.5) is of Bernoulli type and readily solvable.

[2] In the equation which appears in [17], $\frac{dz_1}{dt} + \frac{1}{T}\left(z_1{}^3 + \frac{1-T}{3t} z_1 + \frac{1}{t}\right) = 0$, the term $z_1{}^3$ should be replaced by $-z_1{}^3/t^2$.

With respect to Abel equations of first kind Eq.(3.5), in [16], Liouville defined the relative invariants s_{2n+3} of weight $2n+3$ and multiplier $\tilde{\varphi} = F'P$:

$$s_3 \equiv f_0 {f_3}^2 + \frac{2{f_2}^3}{27} + \frac{f_3 f_2' - f_2 f_3' - f_1 f_2 f_3}{3}, \quad s_{n+2} \equiv f_3 {s_n}' - n s_n \left({f_3}' + f_1 f_3 - \frac{{f_2}^2}{3} \right).$$

For equations in rational normal form, the corresponding invariants have multiplier $\tilde{\varphi} = {F'}^2$ and can be written more compactly

$$s_3 = {c_3}^2, \qquad s_{2n+3} = \tilde{D}^n s_3, \tag{3.9}$$

where the differential operator $\tilde{D}$ defined by

$$\tilde{D}(z_n) \equiv c_3 {z_n}' - n z_n \left({c_3}' + c_1 c_3 \right)$$

takes invariants z_n of weight n to those of weight $n+2$. He further defined the absolute invariants J_1, J_2, and J_3, ratios of the relative invariants s_n given above:

$$J_1 := \frac{{s_5}^3}{{s_3}^5}, \qquad J_2 := \frac{s_5 s_7}{{s_3}^4}, \qquad J_3 := \frac{s_9}{{s_3}^3}. \tag{3.10}$$

Integration strategies for the case where J_1 is constant were first provided by Abel [1] and Liouville [13]. Appell showed in [4] a simple equivalence transformation bringing the equation into a form with constant coefficients.

When J_1 is not constant we must solve the system Eq.(2.15) relating the invariants of the input and target equations. For this purpose, if we use the standard equivalence method described in Subsection 2.1, we require one more invariant than unknown variable. For a 3-parameter class, this means five invariants total, so we extend the above sequence by two:

$$J_4 := \frac{s_{11}}{{s_3}^2 s_5}, \qquad J_5 := \frac{s_{13}}{s_3 {s_5}^2}. \tag{3.11}$$

With respect to rational normal form equations, a sequence of relative invariants r_n with lesser weights n and a simpler multiplier $\tilde{\varphi} = F'$ can be defined via

$$r_3 \equiv c_3, \qquad r_{n+1} \equiv {c_3}^{n-1} \tilde{D}(r_n {c_3}^{-n}) = n\, c_1 r_n + \frac{d}{dx} r_n, \quad n \geq 3.$$

However, the absolute invariants arising as products of these relative invariants do not seem to be optimal for matching the AIL and AIR classes. This may be because for these classes the solvable equations under consideration are of the AIA class and consequently the coefficients f_i of Eq.(3.5) are polynomials and considerably simpler than the c_i of Eq.(3.6), as we shall see in Section 3.4 below. Nevertheless we will use as a particularly convenient relative invariant the ratio

$$r_1 \equiv \frac{r_4}{r_3} = -\frac{s_5}{{c_3}^3} = 3c_1 + \frac{{c_3}'}{c_3}, \tag{3.12}$$

a quantity which cannot be expressed as ratios of integral powers of the s_n.

We now introduce a new sequence of absolute invariants, the first two of which L_1 and L_2 are the invariants t and T respectively of Eq.(3.8):

$$L_1 \equiv \frac{{s_5}^3}{{s_3}^5}, \quad L_2 \equiv \frac{DL_1}{L_1} = \frac{t_{10}}{{s_5}^2}, \quad L_3 \equiv \frac{Dt_{10}}{t_{10}} = \frac{s_3 t_{12}}{s_5 t_{10}}, \ \ldots, \quad L_n \equiv \frac{Dt_{2n+4}}{t_{2n+4}}, \tag{3.13}$$

where $t_{10} \equiv 3s_3s_7 - 5{s_5}^2, t_{2n+10} \equiv \tilde{D}^n t_{10}$, and[3]

$$D \equiv \frac{s_3}{s_5}\tilde{D}.$$

Note that the differential operator D preserves the weight of an invariant.

To demonstrate the advantage of using these new invariants as opposed to the original sequence based on ratios of powers of the relative invariants s_n, we compare some properties of the two sequences in Tables 3.1 and 3.2. Note that L_2 is a significantly simpler invariant than J_2, with respect to degree or expression size (as measured with Maple). Next, while L_3 appears more complicated than J_3 from the point of view of degree and expression size, note that it is the product of four distinct powers of polynomials as opposed to two for the invariant J_3. This in fact makes it a much more convenient choice for the subsequent decomposition into invariant component polynomials, as described in Chapter 2. Finally, we should note that while five invariants are necessary to solve the equivalence problem using the standard method, only three are necessary using the ICP method. Thus with the new method we avoid calculating J_4 and J_5, which it should be noted are significantly more complicated than J_1, J_2, and J_3.

Table 3.1: Properties of Abel absolute invariants introduced in [13, 16], evaluated at the general AIR representative Eq.(3.23)

Invariant	Formula	Degree in				Maple	# of
		x	d	b	c	length	factors
J_1	$\frac{{s_5}^3}{{s_3}^5}$	30	10	15	10	1259	2
J_2	$\frac{s_5 s_7}{{s_3}^4}$	24	8	12	8	4586	3
J_3	$\frac{s_9}{{s_3}^3}$	18	6	9	6	8567	2
J_4	$\frac{s_{11}}{{s_3}^2 s_5}$	22	6	11	6	18655	3
J_5	$\frac{s_{13}}{s_3 {s_5}^2}$	26	8	13	8	35716	2

[3] Using the notation in [5], we have $D = \frac{J}{J'(X)}\frac{d}{dX} = \frac{d}{d\ln(J)}$.

Table 3.2: Properties of Abel absolute invariants in Eq.(3.13), evaluated at the general AIR representative Eq.(3.23)

Invariant	Formula	Degree in				Maple length	# of factors
		x	d	b	c		
L_1	$\frac{{s_5}^3}{{s_3}^5}$	30	10	15	10	1259	2
L_2	$\frac{DL_1}{L_1}$	20	6	10	6	2816	3
L_3	$\frac{Dt_{10}}{t_{10}}$	18	7	11	7	8847	4

3.3 Minimal representatives

In the work [9] we showed that the AIL super-class can be split into a set of subclasses depending on at most two parameters; similarly in [7, 22] it was shown that the AIR super-class can be split into a set of subclasses depending on at most three parameters. In this section we will refine these classifications, splitting AIL and AIR into sets of minimal subclasses depending on at most two and three parameters respectively. Minimal representative equations for these subclasses will then be used as target equations for an ICP matching algorithm for the complete set of AIL and AIR equations.

3.3.1 AIL

We start by reducing the general AIL Eq.(3.3) representative to a single equivalent representative depending on four parameters.

Proposition 3.3.1. *Every equation of the form*

$$y' = \frac{a_3 y^3 + a_2 y^2 + a_1 y + a_0}{(s_1 x + s_0)y + r_1 x + r_0}$$

not equivalent to a separable equation is equivalent to an equation of the form:

$$AIL_{d,a,b,c} : y' = -\frac{ay^3 + dy^2 + by + c}{y - x}. \tag{3.14}$$

Proof. Since the equation is not separable, at least one of s_1 and r_1 is not 0. If $s_1 \neq 0$, we can achieve an equation with $s_1 = 0$ and $r_1 \neq 0$ via the transformation $y \mapsto \frac{1}{y} - \frac{r_1}{s_1}$. So we can assume $s_1 = 0$, $r_1 \neq 0$, and now the condition of non-separability further implies that $s_0 \neq 0$. At this point an affine transformation in either x or y suffices to remove r_0 and make $s_0 = 1$ and $r_1 = -1$. For brevity, we rename the constant coefficients in the numerator of the right hand side, as they will play a prominent role in the following development. □

We now split the class represented by Eq.(3.14) into minimal subclasses. First note that the equation with $a = d = 0$ has a constant invariant. If $a = 0$, $d \neq 0$, a transformation in x and y can remove the coefficient b, and make $d = 1$:

$$y' = -\frac{y^2 + c}{y - x}. \tag{3.15}$$

If $c = -\frac{1}{4}$, this equation has constant invariant.

If $a \neq 0$, a transformation in x and y can remove the coefficient d, and make either $c = 1$ if $c \neq 0$, or $a = 1$ if $c = 0$:

$$y' = -\frac{ay^3 + by + 1}{y - x}, \tag{3.16}$$

$$y' = -\frac{y^3 + by}{y - x}.$$

The first equation has a constant invariant if $a = -\frac{1}{27}(b+1)(2b-1)^2$. The latter equation is invariant under the symmetry $(x, y) \mapsto (-x, -y)$, and thus by Theorem 2.3.5 not minimal; a minimal representative is obtained via the transformation $(x, y) \mapsto (\sqrt{x}, \sqrt{x}\, y)$:

$$y' = -\frac{1}{2}\frac{y(y - 1 + y^2 x + b)}{x(y - 1)}. \tag{3.17}$$

Theorem 3.3.2. *Every nonseparable Abel inverse linear equation with nonconstant invariant is equivalent to Eq.(3.15) with $c \neq -\frac{1}{4}$, Eq.(3.16) with $a \neq -\frac{1}{27}(b + 1)(2b - 1)^2$, or Eq.(3.17).*

We also claim that all these equations have nonconstant invariant and are minimal. The proofs will be that we can demonstrate an ICP algorithm for them.

3.3.2 AIR

As with AIL, we begin by reducing the general AIR representative to an equivalent one depending on four parameters.

Proposition 3.3.3. *Every equation of the form*

$$y' = \frac{a_3 y^3 + a_2 y^2 + a_1 y + a_0}{(s_2 x^2 + s_1 x + s_0)y + r_2 x^2 + r_1 x + r_0} \tag{3.18}$$

is equivalent to either a separable or inverse-linear equation, or an equation of the form:

$$AIR_{d,a,b,c} : y' = -\frac{dy^3 + ay^2 + by + c}{y - x^2}. \tag{3.19}$$

Proof. The first step is to arrive at a representative with $r_2 = r_1 = 0$:

$$y' = \frac{a_3 y^3 + a_2 y^2 + a_1 y + a_0}{(s_2 x^2 + s_1 x + s_0)y + r_0}. \tag{3.20}$$

We start by applying to Eq.(3.18) the linear transformation $y \mapsto y + A$:

$$y' = \frac{a_3y^3 + \tilde{a}_2y^2 + \tilde{a}_1y + \tilde{a}_0}{(s_2x^2 + s_1x + s_0)y + (r_2 + As_2)x^2 + (r_1 + As_1)x + (r_0 + As_0)}. \tag{3.21}$$

The discriminant of $(r_2+As_2)x^2+(r_1+As_1)x+(r_0+As_0)$ with respect to x can be solved for A unless

$$s_1^2 = 4s_2s_0, \qquad r_1s_1 = 2(s_2r_0 + r_2s_0).$$

This condition implies that Eq.(3.18) is either separable ($s_2 = s_1 = s_0 = 0$) or a member of AIL: if $s_2 = s_1 = r_2 = 0$ this is apparent; if $s_1 = s_0 = r_0 = 0$ we must apply $x \mapsto 1/x$; otherwise $s_1 \neq 0$ and we use $x \mapsto 1/x - 2s_0/s_1$. When A does solve the discriminant, Eq.(3.21) takes on either the form Eq.(3.20), or else

$$y' = \frac{a_3y^3 + a_2y^2 + a_1y + a_0}{(s_2x^2 + s_1x + s_0)y + r_2(x - r_0)^2},$$

which takes the form Eq.(3.20) after $x \mapsto 1/x + r_0$.

If $s_2 = 0$, Eq.(3.20) represents an inverse-linear equation; if $r_0 = 0$ it is separable. Otherwise, the transformation $y \mapsto 1/y$ followed by appropriate affine transformations in x and y yields Eq.(3.19). □

We now split Eq.(3.19) into minimal subclasses. By scaling the variables with $(x, y) \mapsto (\xi x, \eta y)$, and if necessary using an extra transformation of the form $(x, y) \mapsto (-1/x, 1/y)$, Eq.(3.19) can be reduced to one of two canonical forms:

$$AIR[c] : y' = -\frac{y^3 + c}{y - x^2}. \tag{3.22}$$

$$AIR[d, b, c] : y' = -\frac{dy^3 + y^2 + by + c}{y - x^2}; \tag{3.23}$$

Note that Eq.(3.22) is unchanged by the transformation $(x, y) \mapsto (\zeta x, \zeta^2 y)$, where $\zeta^3 = 1$. When $c \neq 0$, it is also unchanged by $(x, y) \mapsto (-z/x, z^2/y)$, where $z^3 = c$. By Theorem 2.3.5, Eq.(3.22) is not minimal; we apply the transformation

$$\left(x^3 - \frac{c}{x^3}, \frac{x^4y + c}{2x(x^2 - y)}\right) \mapsto (x, y)$$

to obtain a minimal representative:

$$\tilde{y}' = \frac{8\tilde{y}^3 - 8\tilde{y}^2 + (2\tilde{x} + 6c)\tilde{y} + c\tilde{x} - 4c}{6\tilde{x}^2 + 24c}. \tag{3.24}$$

When $b \neq 0$, the transformation

$$(x, y) \mapsto \left(-\frac{b}{x}, \frac{b^2}{y}\right) \tag{3.25}$$

manifests an equivalence between equations $AIR[d, b, c]$ and $AIR[cb^{-3}, b, db^3]$. In order to achieve a one to one correspondence between triples and classes in the general case, we can instead identify the class by the triple $[b,\ cd,\ c + db^3]$, since the system

$$\{b = \tilde{b},\quad cd = \tilde{c}\tilde{d},\quad c + db^3 = \tilde{c} + \tilde{d}\tilde{b}^3\}$$

has just one nontrivial solution,

$$\{b = \tilde{b}, c = \tilde{d}\tilde{b}^3, d = \tilde{c}\tilde{b}^{-3}\},$$

which preserves the class upon substitution into Eq.(3.23). When $b \neq 0$, we therefore use the notation $AIR\{b, cd, c + db^3\}$ to denote the class represented by Eq.(3.23).

When $c = db^3$, the transformation Eq.(3.25) is a symmetry of Eq.(3.23), which is therefore not minimal. When d is also nonzero, we introduce the new parameters

$$A = 1/bd, \quad C = b^3 d^2$$

and apply the transformation $x \mapsto \sqrt{-C}Ax, y \mapsto -CA^2 y$, obtaining

$$y' = \sqrt{-C}\frac{(y-1)(y^2 + (1-A)y + 1)}{y - x^2}, \tag{3.26}$$

which is invariant under $x \mapsto x^{-1}, y \mapsto y^{-1}$. Consequently, the further transformation

$$\left(\frac{yx^4 - 1}{yx - x^3}\sqrt{-C}, \left(x + \frac{1}{x}\right)\sqrt{-C}\right) \mapsto (x, y)$$

takes the equation to the form

$$\begin{aligned}(x^2 + 4C)(x^2 + C)^2 y' = {}& (Cx^2 + C^2(A+1))\, y^3 \\ &+ (CAx^3 - 2Cx^2 + 4C^2Ax - 2C^2)\, y^2 \\ &+ (x^5 + CAx^4 + 4Cx^3 + 3C^2(A+1)x^2 + 3C^2x - C^3(A-3))\, y \\ &+ Cx^5 + 4C^2x^3 - 4C^2x^2 - C^3(A-3)x - 4C^3, \quad A, C \neq 0.\end{aligned} \tag{3.27}$$

We will show that this equation is minimal except when $(A, C) = (-5, -1/64)$ or $(-5, 1/16)$.

When $(A, C) = (-5, -1/64)$, we start from Eq.(3.27), apply $4x + 1/4x \mapsto x$ and convert to rational normal form, obtaining the minimal representative

$$y' = -\frac{25(5x-8)^2}{729x^3(x+2)^2(x-2)^3}y^3 + \frac{60x^3 - 89x^2 - 128x + 256}{6x(x-2)(5x-8)(x+2)}y + 1. \tag{3.28}$$

Similarly, when $(A, C) = (-5, 1/16)$, we apply $2x + 1/2x \mapsto x$ and convert to rational normal form, obtaining the minimal equation

$$y' = \frac{50}{729x^3(x+2)^3}y^3 + \frac{2(3x^2 - 7x - 7)}{3x(x-2)(x+2)}y + 1. \tag{3.29}$$

For $AIR[0, b, 0]$ with $b \neq 0$ we start from Eq.(3.23), apply $x - b/x \mapsto x$, convert to rational normal form, and finally apply $y \mapsto 9y(x^2 + 4b)/(2x^3 + 6x^2 + (9b + 6)x + 18b - 16)$, arriving at:

$$y' = \frac{y^3 + (-3x^2 + 3x - 9b - 12)y + 2x^3 + 6x^2 + (9b + 6)x + 18b - 16}{9x^2 + 36b}, b \neq 0. \tag{3.30}$$

We have proved:

Theorem 3.3.4. *Every Abel Inverse Riccati equation is equivalent to either a separable or inverse-linear equation; Eq.(3.23) with $b = 0$ or $c \neq db^3$; Eq.(3.24); Eq.(3.27) with $A \neq 0$, $C \neq 0$ and $(A, C) \notin \{(-5, \frac{1}{16}), (-5, -\frac{1}{64})\}$; or one of Eqs.(3.28-3.30).*

Furthermore, we claim that each of these equations, has a non-constant invariant and is minimal. The proofs will be that we can demonstrate an ICP algorithm for them.

3.4 Invariants and ICPs for AIL and AIR classes

In this section we will develop formulae for the ICPs of hypothetical input ODEs representing each of the AIL and AIR super-classes. To do this, we will start with the corresponding representative equation (3.14) or (3.19) and then apply a general rational transformation $x \mapsto F(x)$, where $F = G/H$ with G and H relatively prime polynomials, before converting to rational normal form. We compute the invariants as in Section 3.2 and then the ICPs as in Section 2.2.

These formulae are useful for two reasons. Firstly, we use them to develop and explain the ICP equivalence algorithms, to be presented in the next section, for the minimal subclasses determined in Section 3.3. Secondly, in Section 3.6 we will use them to help determine the subclasses of these minimal classes having cancelled factors.

As a convention, we use lowercase variables to denote newly introduced intermediate polynomials in F, with the subscript signifying the degree (for general values of the parameters). The prime $'$, except as it occurs on y, F or c_3, will denote a derivative with respect to F. Derivatives with respect to x will be denoted by a subscript x. The ICPs will be homogeneous polynomials in G and H; for convenience we will give them uppercase names with a subscript denoting the degree in G and H.

3.4.1 AIL

We start by transforming equation Eq.(3.14) using $x \mapsto F(x)$, using the notation $q(y) = ay^3 + dy^2 + by + c$:

$$\frac{y'}{F'} = -\frac{q(y)}{y - F}. \tag{3.31}$$

Beginning the conversion to rational normal form by applying the transformation $y \mapsto 1/y + F(x)$, we get

$$\frac{y'}{y^2} - F_x = q\left(\frac{1}{y} + F\right) yF',$$

or, upon expanding q in series about $1/y$ and solving for y',

$$y' = (q_3 y^3 + 3j_2 y^2 + j_1 y + a)F',$$

where $q_3 \equiv q(F)$, $j_2 \equiv \frac{1}{3}(1 + q_3{}')$, $j_1 \equiv \frac{1}{2}q_3{}''$. Applying the further transformation $y \mapsto y - j_2/q_3$ removes the coefficient of y^2, yielding

$$y' = \left(q_3 y^3 + \frac{k_2}{q_3} y + \frac{p_3}{q_3{}^2}\right) F'$$

where

$$k_2 \equiv j_1 q_3 - 3j_2{}^2, \tag{3.32}$$

$$p_3 \equiv j_2{}' q_3 - j_2 q_3{}' - j_1 q_3 j_2 + a q_3{}^2 + 2j_2{}^3. \tag{3.33}$$

A final transformation, $y \mapsto \frac{p_3}{q_3{}^2} y$, brings the equation into rational normal form Eq.(3.6) with

$$c_3 = \frac{p_3{}^2}{q_3{}^3} F'^3, \quad c_1 = \frac{k_2 p_3 + 2q_3{}' p_3 - p_3{}' q_3}{p_3 q_3} F' - \frac{F''}{F'}.$$

We use (3.12, 3.13) to calculate the relative and absolute invariants for Eq.(3.19):

$$r_1 \equiv 3c_1 + \frac{c_3{}'}{c_3} = \left(\frac{3k_2 + 3q_3{}'}{q_3} - \frac{p_3{}'}{p_3}\right) F' = \frac{p_5}{q_3 p_3} F', \tag{3.34}$$

$$L_1 = -\frac{p_5{}^3}{p_3{}^5}, \quad L_2 = \frac{k_3 q_3{}^2}{p_5{}^2}, \quad L_3 = \frac{k_5 p_3}{k_3 p_5};$$

where we have defined[4]

$$p_5 \equiv 3p_3(k_2 + q_3{}') - p_3{}' q_3, \tag{3.35}$$

$$k_3 \equiv (5p_3{}' p_5 - 3p_5{}' p_3)/q_3, \tag{3.36}$$

$$k_5 \equiv 2k_3(5k_2 + 4q_3{}') - k_3{}' q_3. \tag{3.37}$$

[4]Note that k_3 is indeed a polynomial - one can check that q_3 divides $5p_3{}'p_5 - 3p_5{}'p_3$ by substituting the expressions for p_5, p_3, and k_2.

If the parameters are general, the invariant component polynomials for the invariant sequence $L = [L_1, L_2, L_3]$ are computed as follows:

$$\begin{aligned} P_3 &:= L_{[-5,0,1]} = \alpha^3 p_3 H^3, \\ P_5 &:= L_{[3,-2,-1]} = \alpha^5 p_5 H^5, \\ Q_7 &:= L_{[0,1,0]} = \gamma {q_3}^2 H^7, \\ K_3 &:= L_{[0,1,-1]} = \frac{\alpha^{10}}{\gamma} k_3 H^3, \\ K_5 &:= L_{[0,0,1]} = \frac{\alpha^{12}}{\gamma} k_5 H^5, \end{aligned} \tag{3.38}$$

where the five unknown constants $\{\alpha_M, M \in S_L\}$ have been reduced to $\{\alpha, \gamma\}$ using the relations

$$L_1 = -\frac{{L_{[3,-2,-1]}}^3}{{L_{[-5,0,1]}}^5}, \quad L_2 = \frac{L_{[0,1,-1]} L_{[0,1,0]}}{{L_{[3,-2,-1]}}^2}, \quad L_3 = \frac{L_{[0,0,1]} L_{[-5,0,1]}}{L_{[0,1,-1]} L_{[3,-2,-1]}}.$$

3.4.2 AIR

As above we transform equation Eq.(3.19) using $x \mapsto F(x)$, using the notation $q_3(y) = dy^3 + ay^2 + by + c$:

$$AIR_{d,a,b,c} : \frac{y'}{F'} = -\frac{q_3(y)}{y - F^2}, \tag{3.39}$$

and then beginning the conversion to rational normal form by applying the transformation $y \mapsto 1/y + F(x)^2$, yielding

$$\frac{y'}{y^2} - 2FF_x = q_3\left(\frac{1}{y} + F^2\right) yF',$$

or, upon expanding q_3 in series about $1/y$ and solving for y',

$$y' = (q_6 y^3 + 3 j_4 y^2 + j_2 y + d)F',$$

where $q_6 \equiv q_3(F^2), j_4 \equiv \frac{1}{3}(2F + q_3{}'(F^2)), j_2 \equiv \frac{1}{2} q_3{}''(F^2)$. Applying the further transformation $y \mapsto y - j_4/q_6$, we obtain

$$y' = \left(q_6 y^3 + \frac{j_5}{q_6} y + \frac{p_6}{{q_6}^2}\right) F',$$

where

$$j_5 \equiv j_2 q_6 - 3{j_4}^2, \tag{3.40}$$

$$p_6 \equiv {j_4}' q_6 - j_4 {q_6}' - j_2 q_6 j_4 + d\, {q_6}^2 + 2{j_4}^3. \tag{3.41}$$

A final transformation, $y \mapsto \frac{p_6}{{q_6}^2} y$, brings the equation into rational normal form Eq.(3.6) with

$$c_3 = \frac{{p_6}^2}{{q_6}^3} F'^3, \quad c_1 = \frac{j_5 p_6 + 2{q_6}' p_6 - {p_6}' q_6}{p_6 q_6} F' - \frac{F''}{F'}.$$

We use (3.12, 3.13) to calculate the relative and absolute invariants for Eq.(3.19):

$$r_1 \equiv 3c_1 + \frac{{c_3}'}{c_3} = \left(\frac{3j_5 + 3{q_6}'}{q_6} - \frac{{p_6}'}{p_6}\right) F' = \frac{p_{10}}{q_6 p_6} F', \tag{3.42}$$

$$L_1 = -\frac{{p_{10}}^3}{{p_6}^5}, \quad L_2 = \frac{p_8 {q_6}^2}{{p_{10}}^2}, \quad L_3 = \frac{p_{12} p_6}{p_8 p_{10}};$$

where we have defined[5]

$$p_{10} \equiv 3p_6(j_5 + {q_6}') - {p_6}' q_6, \tag{3.43}$$

$$p_8 \equiv (5{p_6}' p_{10} - 3p_6 {p_{10}}')/q_6, \tag{3.44}$$

$$p_{12} \equiv 10 p_8 j_5 + 8 p_8 {q_6}' - {p_8}' q_6. \tag{3.45}$$

If the parameters are general, the invariant component polynomials for the invariant sequence $L = [L_1, L_2, L_3]$ are computed as follows:

$$\begin{aligned} Q_{12} &:= L_{[0,1,0]} = \gamma^2 {q_6}^2 H^{12}, \\ P_6 &:= L_{[-5,0,1]} = \alpha^3 p_6 H^6, \\ P_{10} &:= L_{[3,-2,-1]} = \alpha^5 p_{10} H^{10}, \\ P_8 &:= L_{[0,1,-1]} = \frac{\alpha^{10}}{\gamma^2} p_8 H^8, \\ P_{12} &:= L_{[0,0,1]} = \frac{\alpha^{12}}{\gamma^2} p_{12} H^{12}, \end{aligned} \tag{3.46}$$

where the five nonzero constants $\{\alpha_M, M \in S_L\}$ are reduced to $\{\alpha, \gamma\}$ by the relations

$$L_1 = -\frac{{L_{[3,-2,-1]}}^3}{{L_{[-5,0,1}}^5}, \quad L_2 = \frac{L_{[0,1,-1]} L_{[0,1,0]}}{{L_{[3,-2,-1]}}^2}, \quad L_3 = \frac{L_{[0,0,1]} L_{[-5,0,1]}}{L_{[0,1,-1]} L_{[3,-2,-1]}}.$$

3.5 ICP equivalence algorithms

We will now present a set of ICP algorithms, each of which matches an input equation to one or more of the minimal equations described in Section 3.3. Each algorithm starts with the ICPs for the input equation, as computed in the previous section, Section 3.4, and computes a series of pseudo-invariants using the types of steps described in Section 2.2.4, until finding the parameters and equivalence function F. As mentioned therein, after each such step, we must determine for which subclasses the step could possibly fail or return the wrong result, and provide an alternative branch for that case. Throughout each algorithm there are various conditions to check which are necessary for an equivalence. If any of these conditions fail, we return FALSE. Finally, although we do not mention them explicitly, it

[5] Note that p_8 is indeed a polynomial - one can check that q_6 divides $5{p_6}' p_{10} - 3p_6 {p_{10}}'$ by substituting the expressions for p_{10}, p_6, and j_5.

is implicit that there is a verification step as described in section 2.2.4 at the end of each algorithm in this section.

As in the previous section, when the pseudo-invariant has a subscript, it represents the degree of the pseudo-invariant as a homogeneous polynomial in the numerator and denominator G and H of the equivalence function F. The lack of a subscript means the degree is either 0, or unknown. A pseudo-invariant will be labelled as a subscripted R — as in relative invariant — when it is of weight 1; otherwise it is of weight 0.

In this section we will also make use of the following easy to prove lemmas and corollaries:

Lemma 3.5.1. *Suppose the root x_0 has multiplicities m_i in polynomials p_i respectively. Let $n_i \in \mathbb{Z}$. The multiplicity of x_0 in the polynomial*

$$\sum_i n_i {p_i}' \prod_{j \neq i} p_j$$

is:

- $-1 + \sum_{i=1} m_i$ *if* $\sum_{i=1} m_i n_i \neq 0$
- *at least* $\sum_{i=1} m_i$ *if* $\sum_{i=1} m_i n_i = 0$.

Corollary 3.5.2. *Suppose the linear polynomial pseudo-invariant X has multiplicity m_i in polynomial pseudo-invariants P_i respectively, all of weight 0. Let $n_i \in \mathbb{Z}$. The multiplicity of X in*

$$D_* \left(\prod_{i=1} P_i^{n_i} \right) \frac{1}{G'H - GH'}$$

is:

- $-1 + \sum_{i=1} m_i$ *if* $\sum_{i=1} m_i n_i \neq 0$
- *at least* $\sum_{i=1} m_i$ *if* $\sum_{i=1} m_i n_i = 0$.

Lemma 3.5.3. *If A, B, C are polynomials satisfying $A'B - AB' = \delta(A'C - AC')$ for some constant $\delta \neq 0$, then A is proportional to $B - \delta C$.*

Corollary 3.5.4. *If A, B, C are nonzero polynomial pseudo-invariants of weight 0 and all with the same degree, and*

$$D_*(AB^{-1}) = \delta\ D_*(AC^{-1})$$

for some constant $\delta \neq 0$, then A is proportional to $B - \delta C$.

3.5.1 AIL

Algorithm 3.5.1. (Match: $AIL[a,b]$, $a \notin \{0, -\frac{1}{27}(b+1)(2b-1)^2\}$ and $AIR[c]$, $c \neq -1/4$)
Input: The relative invariant r_1 (Eq.(3.12)), and the substituted parent ICPs Eq.(3.38).
Output: $[F,a,b]$, $[F,c]$, or FALSE.

Simplify the weight-1 pseudo-invariant:

$$R_5 := \frac{P_3 Q_7}{P_5} r_1 = \frac{\gamma}{\alpha^2} q_3 H^3 (G'H - GH').$$

Differentiate to obtain a new pseudo-invariant,

$$\begin{aligned} R_6 := D_*(P_3 K_3^{-1}) = & -\frac{15\alpha^{13} a}{\gamma^2}(G'H - GH')\left(27a^2(b+1)G^4 + 162a^2G^3H\right. \\ & + 2(b+1)(8b^3 - 12b^2 - 12b + 27a + 8)GH^3 \\ & \left.-54a(b+1)(-2+b)G^2H^2 + (15b^3 - 36b - 9b^2 + 81a - 12)H^4\right), \end{aligned}$$

and use it to remove $G'H - GH'$ from R_5. The result is

$$Q_3 := \mathrm{denom}\left(\frac{R_6}{R_5}\right) = \beta q_3 H^3$$

for some constant β unless $a \in \{-\frac{4}{27}(b+4)(b-2)^2, -\frac{4}{27}(b-2)(b+1)^2\}$ ($AIR[a,b]$) or $c \in \{-1,-4\}$ ($AIR[c]$), in which case a linear factor cancels from the expression for R_6/R_5. However, we cannot detect this problem immediately. Instead, assuming that Q_3 indeed has the value $\beta q_3 H^3$, we continue by computing

$$\begin{aligned} R_2 &:= \frac{R_5}{Q_3} = \frac{\gamma}{\beta\alpha^2}(G'H - GH'), \\ H_1 &:= \frac{Q_7}{{Q_3}^2} = \frac{\gamma}{\beta^2}H, \\ S_2 &:= \frac{D_*(Q_3 {H_1}^{-3})}{R_2} = \alpha^2 q_3' H^2. \end{aligned}$$

For the class $AIR[c]$, S has the value $2\alpha^2 GH$, so that $S_2/H_1 = 2\frac{\alpha^2\beta^2}{\gamma}G$. In the case $AIR[a,b]$ we compute $G_1 := D_*(S_2 {H_1}^{-2})/R_2 = 6\frac{\alpha^4}{\beta}aG$.

For $AIR[a,b]$ we use Lemma 2.2.12 to find the terms $\frac{\beta^7}{\gamma^3}(aF^3, bF, 1)$ of $Q_3/{H_1}^3$; dividing by the last term gives us aF^3 and bF. If $b=-1$, these two values are sufficient to determine F and a. Otherwise, we obtain $\frac{1}{3}(2b-1)F$ as the ratio of two coefficients $\frac{\alpha^3\beta^6}{\gamma^3}(\frac{1}{3}a(b+1)(2b-1)F^2, a(b+1)F)$ of $P_3/{H_1}^3$; the values of F, b, and a follow straightforwardly. In the case of $AIR[c]$, the terms of $P_3/{H_1}^3$ are $\frac{\alpha^3\beta^6}{\gamma^3}(2F^3, 3F^2, 6(3c+1)F, -(9c+2))$; the first three of which are sufficient to compute the values of F and c.

As noted, the computed value of Q_3 above may be missing a factor linear in F. If so, the above computations will fail at some point. We rename $Q_2 := Q_3$, $H_3 := H_1$, $R_3 := R_2$ and recompute Q_3 via the following computations:

$$\begin{aligned}
Z_2 &:= D_*({Q_2}^3 {H_3}^{-2})/R_3; \\
Z_1 &:= D_*(Q_2 {Z_2}^{-1})/R_3; \\
X_1 &:= \text{denom}(Z_1);\ \text{(here there is no chance for cancelled factors)} \\
Q_3 &:= Q_2 X_1 = \beta q_3 H^3.
\end{aligned}$$

and repeat the above computations with the new value of Q_3.

Algorithm 3.5.2. (Match: $AIL[b]$)
Input: The relative invariant r_1, and the substituted parent ICPs for the minimal class representative Eq.(3.17), under the transformation $x \mapsto G(x)/H(x)$, which are given by

$$\begin{aligned}
H_1 := L_{[-1,0,0]} &= \frac{\alpha_3^3}{\alpha_1^5} H, \\
Q_3 := L_{[0,1,0]} &= \alpha_2 G(Hb+G)^2, \\
P_1 := L_{[-5,0,1]} &= \alpha_1(9G + H(b-2)), \\
P_2 := L_{[3,-2,-1]} &= \alpha_3((9+27b)G^2 - 3H(5+4b+2b^2)G - H^2(b-2)(b^2-b+1)), \\
K_1 := L_{[0,1,-1]} &= \frac{\alpha_3^2}{\alpha_2}((9b+3)G + (b-2)(b-3)H), \\
K_2 := L_{[0,0,1]} &= \frac{\alpha_3^3}{\alpha_1\alpha_2}(9(3b+1)(5b+1)G^2 - 3(2b-1)(5b^2+18b+7)GH \\
&\quad -(b-2)(b-3)(5b^2-2b+5)H^2),
\end{aligned}$$

where $\alpha_1, \alpha_2, \alpha_3$ are three unknown nonzero constants.
Output: $[F, b]$, or FALSE.

We compute:

$$\begin{aligned}
Q_2 &:= \frac{P_2}{r_1}\frac{d}{dx}\ln\left(\frac{P_1}{H_1}\right) = 18\alpha_3 G(G+bH), \\
G_1 &:= \frac{{Q_2}^2}{Q_3} = 324\frac{\alpha_3^2}{\alpha_2} G.
\end{aligned}$$

We now use Lemma 2.2.12 to find the coefficients $9b+3$ and $(b-2)(b-3)F^{-1}$ of the expression K_1/G_1. If $b \notin \{2,3\}$ this is sufficient to find b and F. Under the condition $b=3$ the ICPs $L_{[0,0,1]}$, $L_{[0,1,-1]}$, $L_{[0,1,0]}$ acquire the common factor G; in particular the adjusted $L_{[0,0,1]}$ has degree 1 and we rename it W_1. We can therefore compute

$$\frac{\frac{d}{dx}\ln(H_1 {P_1}^{-1})}{\frac{d}{dx}\ln(W_1 {P_1}^{-1})} = -\frac{3}{175}(48F - 53),$$

from which we obtain F readily. Finally, if $b=2$, we use the fact that $L_3 = \frac{2}{7}\frac{11F-9}{F-1}$ to find F.

3.5.2 AIR

Algorithm 3.5.3. (Match: $AIR[d,b,c]$, $c \neq db^3$, or $c = b = 0$)
Input: The relative invariant r_1, and the substituted parent ICPs Eq.(3.46).
Output: $[F,d,b,c]$, or FALSE.

We start by simplifying the weight-1 pseudo-invariant:

$$R_8 := \frac{r_1 Q_{12} P_6}{P_{10}} = \frac{\gamma^2}{\alpha^2} q_6 H^6 (G'H - GH')$$

The first nontrivial step of this ICP algorithm is again to create a new pseudo-invariant via differentiation. The simplest weight-0, degree-0 pseudo-invariant that we can construct from the given ICPs Eq.(3.46) is $Q_{12}/{P_6}^2$; the corresponding logarithmic derivative is

$$\frac{d}{dx}\ln\left(\frac{Q_{12}}{{P_6}^2}\right) = 2\left(\frac{{q_6}'}{q_6} - \frac{{p_6}'}{p_6}\right)F'.$$

Recalling Eq.(3.42), we notice a useful cancellation by subtracting half of the above expression from the relative invariant r_1:

$$R_0 := r_1 - \frac{1}{2}\frac{d}{dx}\ln\left(\frac{Q_{12}}{{P_6}^2}\right) = \left(\frac{3j_5 + 3{q_6}'}{q_6} - \frac{{p_6}'}{p_6}\right)F' - \left(\frac{{q_6}'}{q_6} - \frac{{p_6}'}{p_6}\right)F' = \frac{3j_5 + 2{q_6}'}{q_6}F'.$$

The new polynomial $3j_5 + 2{q_6}'$ which appears in R_0 can be expressed in terms of q_3 using the definitions for j_5, j_4, and j_2, showing that it is in fact of degree 4, and has only even-powered terms. We call it w_4:

$$\begin{aligned}
w_4 &\equiv 3j_5 + 2{q_6}' \\
&= 3j_2 q_6 - 9{j_4}^2 + 4{q_3}'(F^2)F \\
&= \frac{3}{2}{q_3}''(F^2)q_6 - (2F + {q_3}'(F^2))^2 + 4{q_3}'(F^2)F \\
&= \frac{3}{2}{q_3}''(F^2)q_3(F^2) - {q_3}'(F^2)^2 - 4F^2 \\
&= (3db-1)F^4 + (9dc - b - 4)F^2 + 3c - b^2.
\end{aligned}$$

Multiplying by Q_{12}/R_8 extracts the corresponding pseudo-invariant W_4 from R_0:[6]

$$W_4 := \frac{Q_{12}R_0}{R_8} = \alpha^2 w_4 H^4. \tag{3.47}$$

Starting from Q_{12} and W_4, and using the $*$ operator, we compute a series of pseudo-invariants, which are "sparse", meaning that every second coefficient is zero:

$$\begin{aligned}
W_8 &:= D_*({W_4}^3 {Q_{12}}^{-1})/R_8 = \alpha^4 w_7 H^8, & w_7 &\equiv 3{w_4}'q_6 - 2w_4{q_6}', \\
W_{16} &:= D_*(W_8 {W_4}^{-2})\frac{Q_{12}}{R_8} = \alpha^8 q_6 w_{10} H^{16}, & w_{10} &\equiv {w_7}'w_4 - 2w_7{w_4}' \\
W_{24} &:= \frac{D_*(W_{16}{Q_{12}}^{-1}{W_4}^{-1})}{R_8} = \alpha^{14} q_6 w_{17} H^{24}, & w_{17} &\equiv {w_{10}}'q_6 w_4 - w_{10}{q_6}'w_4 - w_{10}q_6{w_4}'.
\end{aligned} \tag{3.48}$$

[6]Necessary conditions are that W_4 be polynomial in $\tilde{x}$ and not zero.

It is clear that q_6, w_4 and w_{10} have only even powers of F, while w_7 and w_{17} have only odd powers. The GCD of ${W_8}^2$ and ${W_{24}}^2/Q_{12}$ is therefore divisible by G^2H^2, and any extra common factor is of the form $u_n(G^2,H^2)^2$ where u_n is homogeneous of degree $n \leq 3$ in G^2, H^2:

$$U_{4n+4} := \gcd({W_8}^2, {W_{24}}^2/Q_{12}) = G^2H^2u_n(G^2,H^2)^2. \tag{3.49}$$

Note that $W_8 \neq 0$ since $Q_{12}/{W_4}^3$ is never constant.

The algorithm as described up to this point can be used both for subclasses with $c = db^3$, $d, b \neq 0$ and $c \neq db^3$, or $c = b = 0$. From now on, however, we will restrict to the case where Eq.(3.23) is minimal, namely $c \neq db^3$, or $c = b = 0$. We will continue with the subclass $c = db^3$, $d, b \neq 0$ in the next algorithm.

In the following development, note that all computed expressions are sparse homogeneous polynomials in G and H. The polynomials U_{2n+2}, O_k and Z_k all have only odd powers of G and H, while the polynomials E_k have only even powers of G and H. We will also make use of Corollaries 3.5.2 and 3.5.4.

Given that Eq.(3.23) is minimal, we compute the following expressions:

$$Q_6 := \sqrt{Q_{12}} = \gamma q_6 H^6, \quad R_2 := R_8{Q_6}^{-1} = \frac{\gamma}{\alpha^2}(G'H - GH')$$

$$U_{2n+2} := \sqrt{U_{4n+4}} = gcd(W_8, \frac{\alpha^{14}}{\gamma}w_{17}H^{18}) = GHu_n(G^2,H^2).$$

Our aim now is to compute an expression O_2 of the form ρGH for some constant ρ. In the general case, $n = 0$ and we can simply take $O_2 := U_{2n+2}$. When $n = 3$, we show in Appendix A that $u_n(G^2,H^2)$ must have a factor in common with some $Z \in \{Q_6, W_4\}$, and furthermore that G and H divide Z no more often than they do $u_n(G^2,H^2)$. Dividing out a factor common to Q_6 if it exists, or a factor common to W_4 otherwise, we therefore obtain a new $U_{2n+2} = GHu_n(G^2,H^2)$ with $\deg(u_n) = n < 3$:

$$U_{2n+2} := \frac{U_{2n+2}}{\gcd(U_{2n+2}, Q_6)} \quad \text{or} \quad U_{2n+2} := \frac{U_{2n+2}}{\gcd(U_{2n+2}, W_4)}.$$

If $n = 1$ or 2, we proceed by first computing an intermediate expression $E_2 = e_1G^2 + e_2H^2$ if necessary. When $n = 2$, this is straightforward: we assign $E_2 := W_8/U_{2n+2}$. When $n = 1$, we define $E_4 := W_8/U_{2n+2}$, and obtain $O_6 := D_*(E_4{W_4}^{-1})/R_2$, which cannot be zero.[7] If U_{2n+2} does not divide O_6, then we can take O_2 to be their GCD, which is of the form ρGH, and skip finding E_2. Otherwise, we take $E_2 := O_6/U_{2n+2}$. If E_2 divides U_{2n+2}, we take O_2 to be the quotient, skipping the next step. If not, proceed as follows.

[7] If $O_6 = 0$ then W_4 divides W_8. Each factor in W_4 must therefore divide Q_6 using Eq.(3.48) and Corollary 3.5.2. If W_4 were not a square it would divide Q_6 and therefore W_8 twice. So $W_4 = {W_2}^2$ and W_2 divides Q_6, but these conditions imply $c = db^3$.

Having found E_2, compute O_2 as follows. Consider now $Z_6 := D_*(Q_6 {E_2}^{-3})/R_2$ and $Z_4 := D_*(W_4 {E_2}^{-2})/R_2$, which are both non-zero.[8] If Z_4 divides Z_6 with quotient Z_2, we find $O_2 := D_*(E_2 {Z_2}^{-1})/R_2 \neq 0$;[9] otherwise $O_2 := \gcd(Z_4, Z_6)$.

Given $O_2 = \rho G H$ for some unknown constant ρ, we obtain a differential operator on $\mathbf{C}(F)$:

$$\mathcal{D} := \frac{O_2}{R_2}\frac{d}{dx} = \beta F \frac{d}{dF}, \text{ where } \beta \equiv \frac{\alpha^2 \rho}{\gamma}. \tag{3.50}$$

Looking to determine β, we compute

$$\omega := \frac{W_4}{{O_2}^2} = \frac{\alpha^6}{\gamma^2\beta^2}\left((3db-1)F^2 + \frac{3c-b^2}{F^2} + (9dc-b-4)\right),$$

$$\mathcal{D}^n\omega = \frac{2^n\beta^{n-2}\alpha^6}{\gamma^2}\left((3db-1)F^2 + (-1)^n\frac{3c-b^2}{F^2}\right), \quad n>0.$$

The condition $c \neq db^3$ or $b=0$ implies that $\mathcal{D}^n\omega$ never vanishes. We thus obtain more necessary conditions: for all n, $\mathcal{D}^n\omega$ is not constant and the expression

$$\beta^2 := \frac{\mathcal{D}^{n+2}\omega}{4\mathcal{D}^n\omega}, \quad n = 1,2$$

is constant and does not depend on n. Applying the operator $\mathcal{D}$ to $\kappa \equiv Q_6/{O_2}^3$, we obtain:

$$\mathcal{D}^n\kappa = \beta^{n-3}\frac{\alpha^6}{\gamma^2}\left(3^n dF^3 + F + (-1)^n\frac{b}{F} + (-3)^n\frac{c}{F^3}\right), \quad n = 0\,..\,3.$$

We can solve this system for the powers of F if we know β. So we choose a square root of β^2, and call it $\tilde\beta$. If $\tilde\beta = \beta$, we obtain:

$$v := \frac{1}{48}\begin{pmatrix} -3 & -1 & 3 & 1 \\ 27 & 27 & -3 & -3 \\ 27 & -27 & -3 & 3 \\ -3 & 1 & 3 & -1 \end{pmatrix}\begin{pmatrix} \tilde\beta^3\kappa \\ \tilde\beta^2\mathcal{D}\kappa \\ \tilde\beta\mathcal{D}^2\kappa \\ \mathcal{D}^3\kappa \end{pmatrix} = \frac{\alpha^6}{\gamma^2}\begin{pmatrix} dF^3 \\ F \\ bF^{-1} \\ cF^{-3} \end{pmatrix}. \tag{3.51}$$

If $\tilde\beta = -\beta$, the calculated v takes the value $-\alpha^6\gamma^{-2}(cF^{-3}, bF^{-1}, F, dF^3)$. The additional condition $b = 0$ would imply $v_2 = 0$; if this is found to be the case we should choose the other root $\tilde\beta := -\tilde\beta$ and recalculate v as above. Otherwise $b \neq 0$ and so on account of the transformation Eq.(3.25), the function $\tilde F \equiv -b/F$ is an equally valid equivalence function. If we redefine $F := \tilde F$ as the new equivalence function, the definition of β also must change so

[8] It is easy to check that $D_*(Q_6{E_2}^{-3}) = 0$ only when $c = db^3$. In Appendix A we show that Z_4 cannot be zero if $n = 2$. If $n = 1$, $D_*(W_4{E_2}^{-2}) = 0$ means E_2 divides O_6 and W_4, therefore E_4 as well. Therefore E_2 must divide O_6 twice, but this is impossible since E_2 does not divide U_{2n+2}.

[9] If $D_*(E_2{Z_2}^{-1}) = 0$ then E_2 divides Q_6, and therefore the quotient is a linear combination of W_4 and ${E_2}^2$, by Corollaries 3.5.2 and 3.5.4. It is simple to check that these conditions imply $c = db^3$.

that Eq.(3.50) still holds. Since $\tilde{F}\frac{d}{d\tilde{F}} = -F\frac{d}{dF}$, the required redefinition of β is to multiply it by -1. We now have $\tilde{\beta} = \beta$ after all.

We finally compute

$$\frac{\alpha^6}{\gamma^2} := \frac{2}{\tilde{\beta}\mathcal{D}\omega}\left((3v_1v_3 - {v_2}^2) - (3v_2v_4 - {v_3}^2)\right).$$

Using Eq.(3.51), the equivalence function F and the parameters d, b, c follow directly. Necessary conditions are that d, b, c and $\frac{\alpha^6}{\gamma^2}$ are constant and that $F_x \neq 0$.

Algorithm 3.5.4. (Match: $AIR[d, b, db^3], d, b \neq 0$)

Input: The relative invariant r_1, and the substituted parent ICPs Eq.(3.46) for the equation Eq.(3.27).

Output: $[f, d, b]$, or FALSE.

As we have seen, Eq.(3.23) is not minimal when $c = db^3$ $b, d \neq 0$; a minimal representative for this class is given as Eq.(3.27). Recalling that the transformation to go from Eq.(3.23) to Eq.(3.27) involved replacing $x/A - AC/x$ by x, we see that the invariants L_i for the two equations are related through

$$L_i[3.27]\left(\frac{g}{h}\right) = L_i[3.23]\left(\frac{G}{H}\right)|_{c=db^3, b=CA^2, d=1/CA^3},$$

where $g = G^2 - A^2CH^2, h = AGH$ are relatively prime polynomials and $f \equiv g/h$ is the equivalence function to be determined. In the general case the signature for this subclass remains the same as for the class $AIR[d, b, c]$. By Proposition 4.2 the corresponding invariant polynomials satisfy

$$\alpha_M P_M[3.27](g, h) = P_M[3.23](G, H)|_{c=db^3, b=CA^2, d=1/CA^3}.$$

We now make use of the following lemma which is easy to prove:

Lemma 3.5.5. *The sparsity of a homogeneous polynomial $P(g, h)$ is preserved under the substitution*

$$g = a_1G^2 + a_2H^2, h = a_3GH.$$

Moreover a sparse polynomial $P(g, h)$ has even powers of g (and h) if and only if the substituted result has even powers of G (and H).

It follows from Lemma 3.5.5 that Q_{12} and the W_i must be sparse as polynomials in g, h. Similarly, R_8 can be expressed in terms of g, h as

$$R_8 = \frac{\gamma^2}{\alpha^2}\frac{1}{A^4}\left(\frac{1}{C}g^2 + (A+1)h^2\right)(g'h - gh').$$

However, surprisingly, Q_{12} is no longer square:

$$Q_{12} = \frac{\gamma^2}{A^6}(g^2 + 4Ch^2)\left(\frac{1}{C}g^2 + (1+A)h^2\right)^2.$$

We therefore follow the steps of Algorithm 3.5.3 above until the first computation of U_{4n+4}, stopping before the computation of $\sqrt{Q_{12}}$.

Again using Lemma 3.5.5, W_8 and W_{24} have only odd powers of g and h, and as they are of even degree, are both divisible by gh, so that in general U_{4n+4} is a constant multiple of g^2h^2. The four exceptional cases where ${W_8}^2$ and ${W_{24}}^2/Q_{12}$ have a common factor are determined by setting to 0 their resultant with respect to g^2:

$$A = 3; \quad C = \frac{4}{(A-3)^2}; \quad C = \frac{4}{(A+1)(A-3)^2};$$

$$C = \frac{k}{(A-3)^2}, A = -\frac{3(9k+16)(k-4)}{k(27k+20)}.$$

In each case $(W_8/gh)^2$ actually divides $(W_{24}/gh)^2/Q_{12}$, and there is a simple way to determine the expression $O_4 = (\rho gh)^2$ given above. We omit the details, except to say that for the k-class, the subclass corresponding to $k = -1$ is the same as $AIR[-1/9]$ and therefore matched by that algorithm.

Next, we write the square root of U_8 as

$$U_4 := \sqrt{U_8} = \delta\gamma\alpha^{-1}A^{-2}gh,$$

for some constant δ. We now compute new weight-0, degree-0 pseudo-invariants Ψ, Θ, Φ and differential operators D_f, $\mathcal{D}_f$, introducing another unknown constant β:

$$D_f \equiv \frac{U_8}{R_8}\frac{d}{dx} = \frac{\delta^2 C f^2}{f^2 + (A+1)C}\frac{d}{df},$$

$$\Psi \equiv \frac{W_4}{U_4} = -\frac{\alpha^3}{\delta\gamma}\left(A(A-3)f + \frac{3(A+1)(A-3)C+4}{f}\right),$$

$$\Theta \equiv \frac{\mathrm{denom}(D_f\Psi)}{U_4} = \frac{\beta}{\delta^2 C}\left(f + \frac{C(1+A)}{f}\right),$$

$$\Phi \equiv \frac{Q_{12}}{\Theta^2 {U_4}^3} = \frac{\alpha^3\delta}{\beta^2\gamma}\left(f + \frac{4C}{f}\right),$$

$$\mathcal{D}_f \equiv \Theta D_f = \beta f\frac{d}{df}.$$

We are finally able to determine a series of constants:

$$\begin{aligned}
\beta_2 \quad &:= \Phi^{-1}\mathcal{D}_f{}^2\Phi &&= \beta^2,\\
\lambda \quad &:= \Phi\mathcal{D}_f\Psi - \Psi\mathcal{D}_f\Phi &&= \frac{2\alpha^6}{\beta\gamma^2}(4-(A-3)^2C),\\
\mu \quad &:= \Phi\mathcal{D}_f\Theta - \Theta\mathcal{D}_f\Phi &&= \frac{-2\alpha^3}{\delta\gamma}(A-3),\\
\nu \quad &:= \Psi\mathcal{D}_f\Theta - \Theta\mathcal{D}_f\Psi &&= \frac{2\beta^2\alpha^3}{\delta^3\gamma}\frac{(A+1)(A-3)^2C-4}{C},\\
\psi \quad &:= \beta^2\Psi^2 - (\mathcal{D}_f\Psi)^2 &&= \frac{4\alpha^6\beta^2}{\delta^2\gamma^2}A(A-3)(3(A+1)(A-3)C+4),\\
\theta \quad &:= \beta^2\Theta^2 - (\mathcal{D}_f\Theta)^2 &&= \frac{4\beta^4}{\delta^4}\frac{A+1}{C},\\
\phi \quad &:= \beta^2\Phi^2 - (\mathcal{D}_f\Phi)^2 &&= \frac{16\delta^2\alpha^6}{\beta^2\gamma^2}C.
\end{aligned}$$

The case $A=-1$ is identified by the condition $\theta=0$; we obtain $C=\frac{\beta_2\mu^4}{4\phi\nu^2}$. Otherwise, we form an overdetermined system for A and C:

$$\frac{16\beta_2\mu^2}{\phi\theta} = \frac{(A-3)^2}{A+1}, \frac{16\psi}{\phi\theta} = \frac{A(A-3)(3(A+1)(A-3)C+4)}{A+1},$$

$$\frac{\lambda^2\beta_2}{\phi^2\theta} = \frac{((A-3)^2C-4)^2}{256C(A+1)}, \frac{64\beta_2\nu^2}{\phi\theta^2} = \frac{((A+1)(A-3)^2C-4)^2}{(A+1)^2C}.$$

This system admits a unique solution unless $C=-1/(A-3)^2$ or $C=-16/((A-3)^2(A+1))$. In these two cases the system is invariant under the substitution $A\to(15-A)/(A+1)$, as are the classes of the corresponding equations Eq.(3.27). Consequently either of two solutions for (A,C) is valid, except when the substitution is invalid, that is $A=15$, or when $A=(15-A)/(A+1)$, namely $A=-5$.

In the latter case where $A=-5$ and C is either $\frac{1}{16}$ or $-\frac{1}{64}$, there are two choices for f, which indicates that the equation Eq.(3.27) is not minimal. Minimal equivalents are provided in (3.28, 3.29) and can be matched by the standard procedure for non-parameterized classes. Otherwise, given A and C, we obtain f in one of three manners:

$$A\notin\{3,-5\} : \beta := \frac{\mu(5+A)}{(\beta_2{}^{-1}\mathcal{D}_f\Phi\mathcal{D}_f\Theta-\Phi\Theta)(A-3)}, f := \frac{\Psi+\beta^{-1}\mathcal{D}_f\Psi}{\mu A};$$

$$A=-5, C\notin\left\{\frac{1}{16},-\frac{1}{64}\right\} : f := \frac{16C-1}{4\Psi\mu^{-1}+5\Phi(64C+1)\theta\nu^{-1}};$$

$$A=3 : f := \frac{\theta\mathcal{D}_f\Phi}{4\mathcal{D}_f\Theta\mathcal{D}_f\Psi}.$$

Algorithm 3.5.5. (Match: $AIR[0,b,0], b\neq 0$)
Input: The relative invariant r_1, and the substituted parent ICPs Eq.(3.46) for the equation Eq.(3.30).

Output: $[f, b]$, or FALSE.

Algorithm 3.5.4 showed how to match the class $AIR[d, b, db^3]$ when $d \neq 0$. The situation is similar when $d = 0$, even though the minimal equation for the class, namely (3.30), is not obtained from Eq.(3.27) by setting $d = 0$.

The equivalence function $f = g/h$ to be determined is related to that of $AIR[d, b, c]$, namely $F = G/H$, by means of $g = G^2 - bH^2, h = GH$.

Having computed the correct polynomials Q_{12}, P_6, P_{10}, we follow Algorithm 3.5.3 to compute U_4, and proceed by finding $\Psi, \Theta, \Phi, \lambda, \mu, \nu, \psi, \theta, \phi, A, C$, and f as in Algorithm 3.5.4. The only exceptional cases where U_8 has an extra factor occur with b either 4 or $-20/27$. As in the previous section, there are simple methods to determine the offending factor in these cases, and thereby determine the correct value of U_4 and so continue as normal.

We should find $\theta = 0$, $A = -1$, and $C = \frac{1}{16b}$; we will have calculated a value for f but it will not be the correct equivalence function. If d were not zero, the condition $A = -1$ would signify the class given by $d = -\frac{1}{b}$. In order to distinguish between the two classes $AIR[-\frac{1}{b}, b, -b^2]$ and $AIR[0, b, 0]$, we perform a test, calculating the variable

$$\Upsilon \equiv \frac{16Cf + (3 + 64C)f^{-1}}{\Psi}.$$

Υ will be independent of x for the class $AIR[0, b, 0]$, and will depend on x for the class $AIR[-\frac{1}{b}, b, -b^2]$ unless $b = -\frac{1}{16}$. However, it is easy to check that this last case the class $AIR[16, -\frac{1}{16}, -\frac{1}{256}]$ is equivalent to the class $AIR[0, -1, 0]$, and so there is no harm in assuming that we are matching the case $AIR[0, b, 0]$ when Υ does not depend on x.

To finish, we compute $b := \frac{1}{16C}$ and redefine $f := -\frac{3b+4}{f}$ unless $b = -\frac{4}{3}$, in which case we obtain $f := 2\frac{\Psi}{\mu}$.

Algorithm 3.5.6. (Match: $AIR[c]$)
Input: The relative invariant r_1, and the substituted parent ICPs Eq.(3.46) for the equation Eq.(3.24).
Output: $[f, c]$, or FALSE.

A minimal representative for the class $AIR[c]$ is given as Eq.(3.24). The corresponding invariants $L_i[c]$ and $L_i[26]$ satisfy

$$L_i[26](g/h) = L_i[c](G/H),$$

where $g = G^6 - cH^6, h = G^3H^3$ are relatively prime polynomials and $f \equiv g/h$ is the equivalence function to be determined. For general values of c, the signature for Eq.(3.24),

$$\{(-5, 0, 1), (0, 0, 1), (0, 1, -1), (0, 1, 0), (2, -1, -1), (3, -2, -1)\}$$

differs from that for $AIR[d,b,c]$. In the general case, the four ICPs that we need are given by

$$L_{[2,-1,-1]} = -\frac{\delta^3}{\alpha^{15}} h,$$
$$L_{[0,1,0]} = Q_{12} = \gamma^2(g^2 + 4ch^2),$$
$$L_{[-5,0,1]} = P_6 = \alpha^3 \left(\left(2c + \frac{16}{27} \right) h - \left(c + \frac{2}{3} \right) g \right),$$

$$\begin{aligned} L_{[3,-2,-1]} &= -\frac{\alpha^{15} P_{10} G H}{\delta^2 h} \\ &= \frac{\alpha^{15}}{27\delta^2} \left((243c^2 - 108c - 120)g - (810c^2 + 144c - 64)h \right) \end{aligned} \tag{3.52}$$

for some nonzero constant δ. We then define D_f and a sequence of expressions:

$$D_f \equiv -\frac{L_{[3,-2,-1]} {L_{[2,-1,-1]}}^2}{L_{[-5,0,1]} L_{[0,1,0]}} \frac{1}{r_1} \frac{d}{dx} = -\frac{3\delta^4}{\gamma^2 \alpha^{18}} \frac{d}{df},$$

$$e_1 \equiv \frac{L_{[0,1,0]}}{{L_{[2,-1,-1]}}^2}, \quad e_2 \equiv \frac{L_{[3,-2,-1]}}{L_{[2,-1,-1]}}, \quad e_3 \equiv \frac{L_{[-5,0,1]}}{L_{[2,-1,-1]}}, \quad e_4 \equiv \frac{D_f e_3}{{D_f}^2 e_1},$$
$$e_5 \equiv (D_f e_1)^2 - 2 e_1 {D_f}^2 e_1, \quad e_6 \equiv e_3 - e_4 D_f e_1, \quad e_7 \equiv \frac{D_f e_2}{{D_f}^2 e_1}.$$

Next, c can be found as the unique solution of the system

$$(3c+2)(27c+8)e_5 + 34992 c e_4 e_6 = 0, \quad (81c^2 - 36c - 40)e_6 + (108c + 32)e_7 = 0,$$

unless $c = -1/9$ or $c = -16/9$, in which case either solution is valid, as the two corresponding classes are equivalent. Finally f is given by

$$c \neq -\frac{2}{3} : f = -\frac{(3c+2)^2 D_f e_1}{1944 {e_4}^2}, \qquad c = -\frac{2}{3} : f = -\frac{25}{26244} \frac{e_5 D_f e_1}{{e_3}^2}.$$

3.6 Cancelled factor replacement algorithms

In the previous section we presented ICP algorithms for the six minimal AIL or AIR subclasses. These algorithms compute the equivalence function and parameters starting from the ICPs of the class representative. As explained in Chapter 2, the required input for these algorithms is the result of substituting the parameter values defining the class into the ICPs of the generic representative being matched, and if these are no longer relatively prime, they will not be the same as the ICPs computed from the input function itself. Our aim in this section is to develop algorithms which convert the actual ICPs for any input equation into the substituted ICPs for the appropriate minimal subclass, as required for the corresponding ICP algorithm.

Recall some definitions from Chapter 2. The *parent subclass*, in this context, is one of the parameterized minimal subclasses determined in Section 3.3, for which we have defined an algorithm in the previous section. The *child class* is the class of the input equation, a subclass of the parent class. The *parent and child representatives* are representatives of each of these classes such that the child representative is obtained from the parent representative via a parameter substitution. The *parent and child ICPs* are the respective ICPs of these representatives. The *substituted parent ICPs* are the pseudo-invariants obtained from the parent ICPs via the parameter substitution which defines the child class. These expressions may not be the same as the child ICPs, the aim of this section being to compute these substituted parent ICPs from the child ICPs. A *cancelled factor* is a factor which is common to more than one of the substituted parent ICPs (but occurs in at most one of the child ICPs). The *type* $\mathcal{T}_X$ of a linear cancelled factor X is the sequence $e_1, \ldots, e_n$ where X divides the ith substituted parent ICPs e_i times. This sequence depends on how the substituted parent ICPs are ordered; we will be sure to keep a consistent order. Note that a factor is cancelled if its type contains at least two nonzero entries. Knowing the type of all cancelled factors is sufficient to reconstruct the substituted parent ICPs.

For example, consider the AIR class with the ICPs defined in Eq.(3.46) but given the following explicit ordering:

$$[Q_{12}, P_6, P_{10}, P_8, P_{12}].$$

If X divides neither Q_{12} nor P_6 but divides P_{10} three times, P_8 twice, and P_{12} once, we have $\mathcal{T}_X = [0, 0, 3, 2, 1]$, or 00321 for short when there is no possibility of confusion. If the entries of $\mathcal{T}_X$ are all divisible by an integer $n > 0$, we rewrite the type by considering that there are n cancelled factors, all equal to X, instead. So, for example the type 22222 is rewritten as 11111^2. This simplifies the classification of the linear cancelled factors somewhat since if X is a cancelled factor of type 22222 with respect to the class $AIR[d, b, c]$, but $c = db^3$, it may become a cancelled factor of type 11111 with resect to the subclass $AIR[d, b, db^3]$, because of the degree 2 rational transformation that relates the minimal representatives of the respective classes.

We will consider one type T_1 a subtype of another T_2 if the nth entry in T_1 is at least the nth entry in T_2, for each n. Thus, the type 00321 is a subtype of the type 00210. Note that when considering superclasses, both X and the invariant polynomials may depend on parameters. When specializing to a subclass, and in doing so assigning values to these parameters, X might divide one or more of the invariant polynomials more often after the substitution; hence $\mathcal{T}_X$ changes to one of its subtypes. Note that we consider the type 11111^2 to be a subtype of 21100, keeping the above definitions in mind.

We can use the following algorithm to replace a cancelled factor whose type is known:

Algorithm 3.6.1. (Replace cancelled factor with given type)
Where the type $\mathcal{T}_X$ of a cancelled factor X with respect to substituted parent ICPs $P_1, \ldots, P_n$ is $\mathcal{T}_X = [t_1, \ldots, t_n]$, redefine the working ICPs using

$$\mathcal{P}_i := \mathcal{P}_i X^{t_i} \text{ for each } i.$$

We call the *type of a class* to be the product of the types of its linear cancelled factors. For example, consider a class with two cancelled factors, the first, $X = G^2 + AGH + BH^2$ quadratic in G, H of type 10011, and the second, $Y = G + CH$ linear of type 10022; the type of the class is denoted $[10011^2, 10022]$. The required algorithm for transforming the child ICPs for a particular subclass to the substituted parent ICPs for the corresponding minimal superclass is usually dependent just on the type of the subclass with respect to the minimal superclass.

In the remainder of this section we will show how to determine the type of each cancelled factor for an input equation and thereby reconstruct the correct expressions for the substituted parent ICPs. We start by listing the possible cancelled factor types for the various minimal subclasses of the AIL and AIR superclasses.

The general strategy will be to first determine all possible types of cancelled factors for a given minimal class, and then, for a given set of child ICPs, apply the following series of steps to replace the cancelled factors:

Algorithm 3.6.2. (Replace cancelled factors)

1. Initialize the working ICPs to the corresponding values of the (child) ICPs for the input equation: $\mathcal{P}_i := C_i$. For any multiplicity occurring in the signature of the parent invariants but missing from the signature of the child invariants, create a working ICP of the corresponding multiplicity and initialize its value to 1: $\mathcal{P}_i := 1$.

2. Use a lookup table to determine those cancelled factors whose types are uniquely determined by their multiplicity. For each one, apply Algorithm 3.6.1.

3. For each remaining child ICP with multiplicity not among the multiplicities of the parent class, iterate through the candidate types for cancelled factors with non-unique multiplicities, testing each candidate by applying Algorithm 3.6.1 and continuing. If at any point the algorithm fails, continue with the next possible candidate type.

4. Use an algorithm (to be determined below) to replace products of cancelled factors with the same multiplicity (not one of the parent multiplicities) but different types.

5. Use an algorithm (to be determined below) to replace cancelled factors whose multiplicities coincide with the standard multiplicities of the ICPs of the parent class.

6. Multiply the candidate ICPs by appropriate constants so that (3.38) or (3.46) still holds.

3.6.1 Cancelled factor type classification

In this subsection we determine the possible types of a cancelled linear factor for each of the minimal subclasses of AIL and AIR. In this development we will use the following specialization of Lemma 3.5.1:

Lemma 3.6.1. *Suppose the root x_0 has multiplicity m, n in polynomials p, q respectively. The multiplicity of x_0 in the polynomial*

$$M\,p'\,q + N\,p\,q'$$

is:

- $m + n - 1$ *if* $mM + nN \neq 0$
- *at least* $m + n$ *if* $mM + nN = 0$.

AIL

In this subsection we categorize the parameter values for which there is a linear cancelled factor X with respect to the five ICPs $[Q_7, P_3, P_5, K_3, K_5]$, where (recalling Eq.(3.38))

$$Q_7 = L_{[0,1,0]} = \gamma {q_3}^2 H^7, \quad P_3 = L_{[-5,0,1]} = \alpha^3 p_3 H^3, \quad P_5 = L_{[3,-2,-1]} = \alpha^5 p_5 H^5,$$

$$K_3 = L_{[0,1,0]} = \frac{\alpha^{10}}{\gamma} k_3 H^3, \quad K_5 = L_{[0,0,1]} = \frac{\alpha^{12}}{\gamma} k_5 H^5.$$

That is, X divides at least two of these polynomials. We approach this by considering each possible pair of ICPs in order, that is: X has type 11000 and subtypes, then 10100 and subtypes not previously considered, etc. until finally we consider type 00011 and subtypes not previously considered. For each pair we take the resultant between the two polynomials to find a condition on the parameters a, b such that both ICPs under consideration have a common factor. When such a condition is found, we use the associated parameter values to determine how many times the given factor divides each of the ICPs to find its type.

For example, to find when Q_7 and P_3 have a common factor, we first notice that we can write $Q_7 = H{Q_3}^2$ where $Q_3 = q_3 H^3$. H cannot divide P_3 since the coefficient of G^3 in P_3 is $-a^2$, and we have the condition $a \neq 0$. The resultant with respect to F of q_3 and p_3 (recall Eq.(3.33) defining p_3 in terms of q_3 and the definition $q_3 = aF^3 + dF^2 + bF + c$) is

$$\frac{a^3(1 + 4b^3 + 27a - 3b)(216a - 1 - 6b + 32b^3)(27a - 8 - 12b + 4b^3)}{19683}.$$

The first two factors cannot vanish for this class; the second two lead to two of the conditions which we list in Table 3.3. Considering the first of these, $a = -\frac{4}{27}(b-2)(b+1)^2$, we substitute it into the ICPs and factor. We detect one cancelled factor $X = (2+2b)G + 3H$, which divides each of the ICPs twice. We then equate to 0 the resultant between X and each of the quotients; the only solutions for b are $b \in \{-1, 2\}$, which would imply $a = 0$. Consequently, for no value of b can X divide any of the ICPs more than twice.

We summarize the results in Table 3.3. Note that there are four rows in which the types are parameterized by k. In these cases, we were not able to fully solve and apply the conditions on the parameters. However we were able to somewhat restrict the possible types using Eqs (3.35 - 3.37), which serve as relations between the ICPs:

- X divides P_3 and K_5, but none of the other three ICPs: By Eq.(3.35) no factor divides p_3 more than once if it does not also divide p_5. The possible subtypes are $0100k$, $1 \le k \le 5$.

- X divides the two ICPs P_5 and K_3, but neither Q_7 nor P_3: Using (3.36) and (3.37), we find that X must divide P_5 once more than it divides K_3, and K_3 once more than K_5. The possible subtypes are $[0, 0, k, k-1, k-2]$; $k = 2, 3, 4$.

- X divides only the two ICPs P_5, K_5: By Eq.(3.36) X cannot divide P_5 more than once as it does not divide K_3. The possible types are $0010k$; $1 \le k \le 5$.

- X divides only the two ICPs K_3, K_5: By Eq.(3.37), it must divide K_3 once more than it divides K_5. The possible subtypes are therefore $[0, 0, 0, k, k-1]$, $k = 2, 3$.

Concerning Algorithm 3.6.2, only steps 4 and 5 require further analysis. Regarding the former, note that of the possible AIL types, only two lead to the same multiplicity: 11111 and 21100, the multiplicity being $[-1, 0, 0]$. The conditions for cancelled factors of both of these types are obtained by intersecting the conditions for the two classes individually, as found in Table 3.3. This leads to class $AIL[9/250, -7/10]$. In this case we have

$$L_{[-1,0,0]} \simeq X^4 Y^2, \text{ where } X = G + 5H, \quad Y = 3G - 10H.$$

We obtain

$$X := \operatorname{denom}\left(\frac{L_{[3,-2,-1]} L_{[0,1,-1]}}{r_1 L_{[-5,0,1]}} \frac{d}{dx} \ln\left(\frac{L_{[0,1,0]}}{{L_{[0,1,-1]}}^3}\right)\right) \simeq G + 5H$$

and $Y := \sqrt{L_{[-1,0,0]} X^{-2}}$ follows.

Finally, for AIL there is only one possible type of cancelled factor with a multiplicity coinciding with one of the standard multiplicities, namely 20011, ie. multiplicity $3 \times [0, 1, 0]$,

Table 3.3: Cancelled factor types for $AIL[a,b]$, $AIL[c]$, $AIR[0,b,0]$

Factor type wrt. ICPs L_M, $M = \dots$ $[0,1,0]$	$[-5,0,1]$	$[3,-2,-1]$	$[0,1,-1]$	$[0,0,1]$	Conditions on $AIL[a,b]$, $AIL[c]$, $AIR[0,b,0]$ (Default is $AIL[a,b]$, $AIL[c]$)
2	1	1	0	0	$a = -\frac{1}{216}(2b-1)(1+4b)^2$; $c = -\frac{1}{16}$
2	0	0	1	1	$a = -\frac{1}{27}(b-3)(2b+3)^2$; $c = -\frac{9}{4}$
2	0	0	0	1	$(3375a + 71 + 345b + 500b^3)^2 = -216(-7+10b)^2$; $400c^2 - 184c + 25 = 0$
1	1	1	1	1	$a = -\frac{4}{27}(b-2)(b+1)^2$; $c = -1$; $AIR[0,b,0]$: $b = -16, -4, -1$
1	0	1	0	0	$(54a + 2 + 8b^3 + 3b)^2 = -27b^2$; $16c^2 - 4c + 1 = 0$; $b = 0$; $AIR[0,b,0]$: $b^2 - 4b + 16 = 0$
1	0	0	1	1	$AIR[0,b,0]$: $b = -36$, $b^2 + 3b + 1 = 0$
1	0	0	0	2	$(a,b) = \left(\frac{7}{1500}, -\frac{1}{10}\right)$
1	0	0	0	1	$b = -1/10$; $AIR[b]$, $25b^2 - 184b + 400$
0	2	1	2	1	$AIR[b]$, $2b^3 + 3b^2 - 12b + 12 = 0$
0	1	0	0	k	$AIL[a,b]$, $1 \le k \le 5$
0	1	0	0	1	$6144c^3 + 1176c^2 - 239c - 131 = 0$; $AIR[0,b_6,0]$:
0	0	k	$k-1$	$k-2$	$AIL[a,b]$, $k = 2,3,4$
0	0	2	1	0	$3888c^3 - 6588c^2 + 4727c - 5377 = 0$; $AIR[0,b_{10},0]$
0	0	1	0	k	$AIL[a,b]$, $k \ge 1$
0	0	1	0	1	$80621568c^7 - 341381952c^6 + 629831376c^5 - 932910980c^4 + 792990676c^3 - 404071221c^2 + 75396265c - 5631482 = 0$; $AIR[0,b_{14},0]$:
0	0	0	k	$k-1$	$AIL[a,b]$, $k = 2,3$
0	0	0	2	1	$243c^2 + 459c - 127 = 0$; $AIR[0,b_{12},0]$

Table 3.4: Minimal polynomials of b values defining $AIR[0, b, 0]$ classes from Table 3.3

Value	Minimal polynomial
b_6	$192b^6 + 582b^5 + 581b^4 + 10070b^3 + 8244b^2 - 43288b + 49472$
b_{10}	$486b^{10} - 3915b^9 + 11200b^8 + 2308840b^7 + 12350640b^6 + 1621312b^5$ $-44517632b^4 + 39800832b^3 + 20708352b^2 - 21491712b + 64475136$
b_{12}	$192b^6 + 582b^5 + 581b^4 + 10070b^3 + 8244b^2 - 43288b + 49472$
b_{14}	$157464b^{14} - 6987465b^{13} + 122117787b^{12} + 1039359288b^{11} + 1131505472b^{10}$ $-51066094208b^9 + 1206232955072b^8 + 10353433043200b^7 + 9296135378432b^6$ $-33015339585536b^5 + 121129261113792b^4 + 188351049072648b^3$ $+264016107601928b^2 - 6912732561408b + 55301860491264$

the condition being $a = -\frac{1}{27}(b-3)(2b+3)^2$. In this case, $X \simeq (2b+3)G + 3H$ is linear, and for no subclass can there be any extra cancelled factors of this type. We can assume X is the only cancelled factor that has not yet been replaced. To match these classes, note that the working values of the ICPs $L_{[0,0,1]}$ and $L_{[0,1,-1]}$ have degrees 4 and 2 respectively. We compute

$$T := \operatorname{denom}\left(D_*(L_{[0,0,1]}L_{[0,1,-1]}{}^{-2})\frac{L_{[3,-2,-1]}L_{[0,1,-1]}}{r_1 L_{[-5,0,1]}L_{[0,1,0]}}\right) \tag{3.53}$$

which is divisible by X^2 unless $b = -2027/1671$. In the general case, the numerator of $T^2/L_{[0,1,0]}$ is X; when $b = -2027/1671$, the GCD of T and $L_{[0,1,0]}/T^2$ is X.

AIL[**b**]

The class $AIL[b]$ has an extra cancelled factor with multiplicity $[-1, 0, 0]$ in the general case; we list the subtypes and associated conditions in Table 3.5. No subclass has more than one cancelled factor. So step 4 of Algorithm 3.6.2 does not apply. With regards to step 5, the types that give rise to one of the standard multiplicities are: 211000, 200110, 111110, 100110. For each of the corresponding conditions, $b = -1/4, -3/2, 3$, the ICPs include a triple of relatively prime linear pseudo-invariants, so we can use Lemma 2.2.13 to find F. When $b = 2$ we obtain $F := (7L_3 - 18)/(7L_3 - 22)$.

AIL[**c**]

For the class $AIL[c]$, we list the subtypes and associated conditions in Table 3.3. Step 4 of Algorithm 3.6.2 again does not apply. There is just one possible type which gives rise to one of the standard multiplicities: 20011, when $c = -9/4$. In that case, the expression T computed in Eq.(3.53) above is also divisible by X^2, and we again obtain $X \simeq T^2/L_{[0,1,0]}$.

Table 3.5: Cancelled factor types for $AIL[b]$

Factor type wrt. ICPs L_M, $M = \ldots$						Conditions
$[0,1,0]$	$[-5,0,1]$	$[3,-2,-1]$	$[0,1,-1]$	$[0,0,1]$	$[-1,0,0]$	
2	1	1	0	0	0	$b = -\frac{1}{4}$
2	0	0	1	1	0	$b = -\frac{3}{2}$
2	0	0	0	1	0	$20b^2 + 4b + 5 = 0$
1	1	1	1	1	0	$b = 2$
1	0	1	0	0	0	$4b^2 + 2b + 1 = 0$; $b^2 - b + 1 = 0$
1	0	0	1	1	0	$b = 3$
1	0	0	0	1	0	$5b^2 - 2b + 5 = 0$
0	1	0	0	1	0	$16b^2 - 9b + 8 = 0$
0	0	2	1	0	0	$4b^2 - 6b + 17 = 0$
0	0	1	1	1	1	$b = -1/3$
0	0	1	0	1	0	$32b^5 - 44b^4 + 242b^3$ $+5b^2 + 298b + 152 = 0$
0	0	0	0	1	1	$b = -1/5$

AIR

We now categorize the parameter values for which the AIR class has a linear cancelled factor X with respect to the five ICPs $[Q_{12}, P_6, P_{10}, P_8, P_{12}]$, where

$$Q_{12} = L_{[0,1,0]}, \quad P_6 = L_{[-5,0,1]}, \quad P_{10} = L_{[0,1,0]}, \quad P_8 = L_{[3,-2,-1]}, \quad P_{12} = L_{[0,0,1]}$$

as in Eq.(3.46). As with the AIL class, we consider each possible pair of ICPs in order, that is: X has type 11000 (and subtypes), then 10100 and subtypes not previously considered, etc. until finally we consider type 00011 and subtypes not previously considered. For now we only attempt to restrict the set of possible types, without determining an explicit parameterization of the defining subclasses. We will perform a more thorough classification in the following subsections as needed for the required sub-algorithms.

For this analysis it is sufficient to use the ICPs resulting from the $AIR[d,b,c]$ representative Eq.(3.23), even though for the subclass $c = db^3$ this representative is not minimal. Indeed, if X is a cancelled factor of degree n for the ICPs of a minimal representative for $AIR[d,b,db^3]$, then it gives rise to a cancelled factor of degree $2n$ for the ICPs of Eq.(3.23). Consequently, we can assume that Q_{12} is a perfect square, and we can compute its root, Q_6, up to constant factor. Furthermore it means that X must divide Q_{12} an even number of times, ie. the first entry of the type of any cancelled factor X will be even.

At this point it is convenient to simplify the discussion slightly by reverting the context from homogeneous polynomials in G and H to single variable polynomials in F. Thus for example, as long as $X \neq H$ (up to a constant), if X divides Q_6 in $\mathbf{K}[G, H]$ then $\Gamma \equiv X/H$ divides $q_6 = Q_6/H^6$ in $\mathbf{K}[F]$.

All possible types resulting from this analysis are listed in Tables 3.7 and 3.8. Note that for those types having all entries even, we have divided each entry by 2 before entering them in the table, to account for the subclass $c = db^3$.

- Subtypes of **11000**: This means that X divides Q_6 and P_6 at least once each. If X divides Q_6 at least twice then a simple calculation shows that either $d = a = 0$ or $b = c = 0$. By means of the symmetry $x \to \tilde{x}^{-1}, y \to \tilde{y}^{-1}$ these cases are one and the same, so we assume $b = c = 0$. Then $\mathcal{T}_X = 83557$ unless $a = 0$, in which case $\mathcal{T}_X = [12, 3, 5, 7, 9]$.

  Let us now assume that X divides Q_6 just once, and P_6 $k \geq 1$ times. We will demonstrate in subsection 3.6.3 that $k \leq 4$; furthermore with the parametrization listed in Table 3.9 it is possible to demonstrate that $k \leq 2$ when $\Gamma \mid j_4$. Using Eq.(3.43) we find that X divides P_{10} at least k times. In fact we will now show that this lower bound is exact. Using Eq.(3.41) we find that Γ divides j_4 $(2{j_4}^2 - {q_6}')$. If Γ divides j_4 then it divides j_5 as well by Eq.(3.40); Lemma 3.5.1 implies that Γ divides $3p_6{q_6}' - {p_6}'q_6$ only k times. Using Eq.(3.43) we find that Γ divides p_{10} just k times. By writing

$$p_{10} \equiv \left(3j_2q_6 - \frac{9}{2}(2{j_4}^2 - {q_6}')\right) p_6 - \frac{3}{2}{q_6}'p_6 - {p_6}'q_6,$$

  and using a similar analysis we see that the same conclusion holds if Γ divides $2{j_4}^2 - {q_6}'$.

  Lemma 3.5.1 and Eq.(3.44) now suffice to show that Γ divides p_8 $2k - 2$ times. The same lemma, now in combination either with Eq.(3.45) if $\Gamma \mid j_5$ (since $k \leq 4$, Γ does not divide p_8 more than 6 times), or if $\Gamma \mid 2{j_4}^2 - {q_6}'$ with the following:

$$p_{12} \equiv 5(2j_2q_6 - 3(2{j_4}^2 - {q_6}'))p_8 - 7{q_6}'p_8 - p_8'q_6,$$

  shows that Γ divides p_{12} $2k - 2$ times as well. In summary, $\mathcal{T}_X$ can be 21100, 22222, 23344, 24466, 83557, or $[12, 3, 5, 7, 9]$.

- Subtypes of **10100** not previously considered: For this type Γ divides q_6 and p_{10} but not p_6. If we suppose $\Gamma^2 \mid q_6$, we find $\Gamma \mid j_5$ by Eq.(3.43), $\Gamma \mid j_4$ by Eq.(3.40) and hence $\Gamma \mid p_6$ by Eq.(3.41), a contradiction. So Γ divides q_6 exactly once. Let k be the number of times that Γ divides p_8. By Eq.(3.44) Γ divides p_{10} $k + 2$ times. By Eq.(3.43) Γ divides $j_5 + {q_6}'$ at least once. Writing

$$p_{12} = 10p_8(j_5 + {q_6}') - {q_6}^{-1}(p_8{q_6}^2)',$$

we see that Γ divides the first term at least $k+1$ times and the second term exactly k times. Hence Γ divides p_{12} k times as well. The types are $[2, 0, k+2, k, k]$. As we will show in subsection 3.6.5, k cannot be more than 2.

- Subtypes of **10010** not previously considered: Since Γ divides q_6 it must divide p_{12} at least as many times as p_8 by Eq.(3.45). We will show in subsection 3.6.5 that the possible types are $200kk$, $k = 1 \ldots 4$; 20012, 20013, 10012^2, 40011.

- Subtypes of **10001** not previously considered: Let k be the number of times Γ divides p_{12}. If $\Gamma^2 \mid q_6$ then $\Gamma \mid j_5$ by Eq.(3.45) and so $\Gamma \mid p_{10}$ by Eq.(3.43), a contradiction. Hence $\Gamma^2 \nmid q_6$. The subtypes are $2000k$, $k \geq 1$.

- Subtypes of **01100** not previously considered: Γ divides p_6 and p_{10} but not q_6. Let k be the number of times Γ divides p_6. From Eq.(3.43) Γ divides p_{10} $k-1$ times. In the context of this subtype we can therefore assume $k \geq 2$. By Eq.(3.44) and Lemma 3.5.1 Γ divides p_8 $2k-2$ times. From Eq.(3.45) Γ divides p_{12} $2k-3$ times. Degree restrictions for p_8 imply $k \leq 5$. We will demonstrate in section 3.6.4 that in fact $k \leq 3$. In summary, we can restrict the possible types to $[0, k, k-1, 2k-2, 2k-3]$, $k = 2, 3$.

- Subtypes of **01010** not previously considered: Impossible: the hypothesis implies that $\Gamma \mid p_6{}'$ by Eq.(3.44), and hence $\Gamma | p_{10}$ by Eq.(3.43), a contradiction.

- Subtypes of **01001** not previously considered: Γ cannot divide p_6 more than once by Eq.(3.43) since it does not divide p_{10}. The possible subtypes are $0100k, k \geq 1$.

- Subtypes of **00110** not previously considered: Using (3.44, 3.45), we find that the only possible subtypes are $[0, 0, k, k-1, k-2]$, $2 \leq k \leq 9$.

- Subtypes of **00101** not previously considered: By Eq.(3.44) Γ cannot divide p_{10} more than once as it does not divide p_8. The types are $0010k, k \geq 1$.

- Subtypes of **00011** not previously considered: By Eq.(3.45), we find that the subtypes are $[0, 0, 0, k, k-1]$.

$\mathbf{AIR}[\mathbf{0}, \mathbf{b}, \mathbf{0}]$, $\mathbf{b} \neq \mathbf{0}$

In general the ICPs for this subclass are again Q_{12}, P_6, P_{10}, P_8, P_{12}. The conditions on b which lead to a cancelled factor X and its type are listed in Table 3.3. The values b_i for $i = 6, 10, 12, 14$ represent the roots of irreducible polynomials of degree i; for brevity we do not display these polynomials.

In all cases there is only one cancelled factor, linear of the given type; consequently step 4 of Algorithm 3.6.2 does not apply. For step 5, note that the only type which leads to one of the standard multiplicities is 10011, under the conditions $b = -36$ or $b^2 + 3b + 1 = 0$. The first class is matched by algorithm 20022: X.i; $AIR[d, b, db^3]$, the second by algorithm [20022: XI.iii; 20422] & $AIR[d, b, db^3]$, both of which will be described below as they are needed for the $AIR[d, b, c]$ algorithm.

AIR[**c**]

For this class there is an extra ICP, H, of multiplicity $[2, -1, -1]$. We list the possible types with respect to the ICPs in Table 3.6.

Table 3.6: Cancelled factor types for $AIR[c]$

Factor type wrt. ICPs L_M, $M = \ldots$						Conditions on c
$[0,1,0]$	$[-5,0,1]$	$[3,-2,-1]$	$[0,1,-1]$	$[0,0,1]$	$[2,-1,-1]$	
1	1	1	1	1	0	$c = -16/9, -4/9, -1/9$
1	0	1	0	0	0	$81c^2 - 36c + 16 = 0$
1	0	0	1	1	0	$c = -4, 81c^2 + 99c + 25 = 0$
1	0	0	0	1	0	$c = -4$, $2025c^2 - 1656c + 400 = 0$
0	1	0	1	1	1	$c = -2/3$
0	1	0	0	1	0	$162c^2 + 63c - 40 = 0$
0	0	1	1	1	1	$81c^2 - 40 - 36c = 0$
0	0	1	0	1	0	$729c^3 - 243c^2 - 144c + 128 = 0$

In the two cases $c = -2/3$ and $81c^2 - 36c - 40 = 0$ the multiplicity $[2, -1, -1]$ is no longer present because the parent ICP now divides one of the others. We detect the signatures for these classes and match them as unparameterized (0-p) classes. For the other values of c, the multiplicity $[2, -1, -1]$ remains in the signature. When we detect it, we exhaust all possibilities that the class could be $AIR[c]$ for some c. Again step 4 does not apply; classes $c = -4$, $c = 0$, and $81c^2 + 99c + 25 = 0$ lead to X being of multiplicity [0,1,0]; for these classes we also develop special 0-p matching algorithms. The other possible types all have non-standard multiplicities, and will be replaced in step 2 or 3 of Algorithm 3.6.2.

3.6.2 Summary of the strategy for replacing cancelled factors.

For the $AIL[a, b]$, $AIL[c]$, $AIR[d, b, c]$ and $AIR[0, b, 0]$ classes we list the possible types $\mathcal{T}_X$ in Tables 3.7 and 3.8, grouping by multiplicity.

The next algorithm combines the various cancelled factor replacement algorithms for the minimal subclasses of AIL and AIR.

Table 3.7: Multiplicities with unique type

Invariant multiplicity	Factor type wrt. ICPs L_M, $M = \dots$ $[0,1,0]$	$[-5,0,1]$	$[3,-2,-1]$	$[0,1,-1]$	$[0,0,1]$	Conditions
$[-5,0,1]$	0	1	0	0	0	
$[-5,0,k+1]$	0	1	0	0	k	$1 \le k \le 12$, $k \ne 4,9$
$[-5,1,1]$	1	1	0	0	0	
$5 \times [-1,0,k]$	0	1	0	0	$5k-1$	$k = 1,2$
$[0,0,1]$	0	0	0	0	1	
$[0,1,-1]$	0	0	0	1	0	
$[0,1,k]$	1	0	0	0	k	$1 \le k \le 6$
$[0,2,k]$	2	0	0	0	k	$k = 3,5,7,9,11$
$[0,k,-1]$	0	0	0	k	$k-1$	$2 \le k \le 8$
$[0,3,1]$	2	0	0	1	2	
$[0,3,2]$	2	0	0	1	3	
$[3,-2,-1]$	0	0	1	0	0	
$[3,-2,k-1]$	0	0	1	0	k	$1 \le k \le 12$
$[3k,-k-1,-k-1]$	0	0	k	$k-1$	$k-2$	$3 \le k \le 9$, $k \ne 5,8$
$3 \times [2,-1,-1]$	0	0	2	1	0	
$3 \times [5,-2,-2]$	0	0	5	4	3	
$3 \times [8,-3,-3]$	0	0	8	7	6	

Algorithm 3.6.3. (Correct invariant polynomials)
Input: The relative invariant r_1, and the ICPs of the input equation.
Output: The substituted parent ICPs with respect to the corresponding minimal class.

1. Verify that all multiplicities in the signature are in one of the two tables above.

2. Apply the various matching algorithms for the 0-parameter subclasses.

3. For each minimal parameterized subclass, apply Algorithm 3.6.2.

Our goal in the following subsections is to provide sub-algorithms for steps 4 and 5 of Algorithm 3.6.2 for the class $AIR[d,b,c]$, that is, to find the types of multiple linear cancelled factors which give rise to the same multiplicity. We start by noting that the two multiplicities which can result from more than one type are $[-1,0,0]$ and $[0,1,0]$. The possible cancelled factor types generating multiplicity $[-1,0,0]$ consist of subtypes of 11000 and 02121 (except 83557 and $[12,3,5,7,9]$). The cancelled factor types generating multiplicity $[0,1,0]$ are all subtypes of 10011 which are not also subtypes of 11000, that is, X divides Q_6 and P_8 but not P_6. For each of these special types we will first determine the parameter restrictions defining the corresponding subclasses.

Table 3.8: Multiplicities with various types

Invariant multiplicity	Factor type wrt. ICPs L_M, $M = \ldots$ $[0,1,0]$	$[-5,0,1]$	$[3,-2,-1]$	$[0,1,-1]$	$[0,0,1]$	Conditions
$6\times[-1,0,0]$	2	3	3	4	4	
$2\times[-1,0,0]$	2	1	1	0	0	
$4\times[-1,0,0]$	1	2	2	3	3	
$2\times[-1,0,0]$	1	1	1	1	1	
$(2k+3)\times[-1,0,0]$	0	k	$k-1$	$2k-2$	$2k-3$	$k=2,3$
$9\times[0,1,0]$	12	3	5	7	9	
$3\times[0,1,0]$	8	3	5	5	7	
$5\times[0,1,0]$	4	0	0	1	1	
$5\times[0,1,0]$	2	0	0	3	3	
$3\times[0,1,0]$	2	0	0	1	1	
$3\times[0,1,0]$	1	0	0	2	2	
$2\times[0,1,0]$	1	0	0	1	1	
$[0,1,0]$	1	0	0	0	0	
$[0,2,1]$	1	0	0	1	2	
$[0,2,1]$	2	0	0	0	1	
$3\times[3,-1,-1]$	2	0	3	1	1	
$2\times[3,-1,-1]$	1	0	2	1	1	
$[3,-1,-1]$	1	0	1	0	0	

3.6.3 Parameter restrictions and cancelled factor algorithms for subtypes of type 11000

Parameter restrictions for subtypes of type 11000

The case $d = c = 0$ has already been dealt with in Algorithm 3.5.5. So, assuming d and c are not both 0, and applying to Eq.(3.19) a scaling $x \to \rho\tilde{x}, y \to \rho^2\tilde{y}$, along with the transformation $x \to 1/\tilde{x}, y \to 1/\tilde{y}$ if $d = 0$, we obtain a representative equation of the form

$$y' = -\frac{(y-A^2)(y-B^2)(y-C^2)}{y-x^2}. \tag{3.54}$$

According to Eq.(3.41), as we have seen in Section 3.6.1, if Γ divides q_6 and p_6 then it also divides $j_4\ (2{j_4}^2 - {q_6}')$, which factors as

$$\frac{2}{27}\prod_{n=-1,2,-4} {q_3}'(F^2) + nF.$$

Since $q_6 = (F^2-A^2)(F^2-B^2)(F^2-C^2)$, we can assume without loss of generality $\Gamma = F - A$, so that for some $n \in \{-1, 2, -4\}$ we have

$$0 = {q_3}'(A^2) + nA = (A^2-B^2)(A^2-C^2) + nA. \tag{3.55}$$

Class III: If $A = 0$ then without loss of generality $B = 0$ as well. The type of the cancelled factor X is 83557 unless $C = 0$ when $\mathcal{T}_X = [12, 3, 5, 7, 9]$, but that is class $AIR[0]$, for which we have already provided an matching algorithm above. There can be an extra cancelled factor Y of type a subtype of 11000 only when $C^3 = \pm n$, $n \in \{-1, 2, -4\}$; $\mathcal{T}_Y = [2, k, k, 2k-2, 2k-2]$, where $k = 1$ if $n = -1$, $k = 2$ otherwise. □

Class V: In the more general case, $A \neq 0$ and therefore $A^2 \neq B^2$. Solving Eq.(3.55) for C^2, substituting into Eq.(3.54) and converting to a more natural representative via $(x, y) \mapsto (Ax, A^2(y+1))$, we obtain the equation for the subclasses defining the subtypes of type 11000:

$$y' = \frac{ny(Wy^2 + (T+1)y + 1)}{y + 1 - x^2}; n = -1, 2, -4, \tag{3.56}$$

where $W = -A^3/n$, and $T = W(1-(B/A)^2)+(1-(B/A)^2)^{-1}-1$. Using this representative, it is straightforward to determine the possible subtypes for the cancelled factor, which is now $X = G - H$. We summarize the subtypes and associated parameter constraints in Table 3.9.

Table 3.9: Subtypes of type 11000 (except 83557 and [12, 3, 5, 7, 9])

Type	Parameter constraints n	T	W
21100	-1		
11111^2	-1	0	
	2		
	-4		
23344	-4		$(16T^2 + 36T + 19)/72$
12233^2	-4	0	19/72

Class VI: Two cancelled factors, both subtypes of 11000. Suppose p_6 and q_6 have another factor Y as well as $X = F - A$ in common. This other factor cannot be $F + A$ since $p_6(-A) = -2(n-1)(n+2)(n-4)A^3/27 \neq 0$. We may assume that the factor is $F - B$. Hence

$$0 = p_6(B) = \frac{2}{27} \prod_{m=-1,2,-4} nA + mB + (A^2 - B^2)^2.$$

Since $A \neq 0$, the condition $B = 0$ would imply $A^3 = -n$ and thus $C = 0$, which would be class III. Therefore $B \neq 0$. Introducing $A_2 = A/B \neq \pm 1$, we solve for B^3:

$$B^3 = -\frac{nA_2 + m}{({A_2}^2 - 1)^2}. \tag{3.57}$$

Applying the transformation $(x, y) \mapsto (Bx, B^2y)$ to Eq.(3.54) we obtain the representatives for six more 1-parameter classes:

$$y' = \frac{(y - {A_2}^2)(y-1)((nA_2+m)y - (mA_2+n)A_2)}{({A_2}^2-1)^2(y-x^2)} \tag{3.58}$$

where $n, m \in \{-1, 2, -4\}, |n| \le |m|, A_2 \ne -m/n$.[10]

We now determine the possible subtypes for the cancelled factors X and Y, and summarize the subtypes and associated parameter constraints in Table 3.10. We have ignored any classes which are also of class $AIR[c]$, as well as the special class $AIR[d, b, db^3]$ with $A = -5, C = 1/16$.

Table 3.10: Factors X and Y, subtypes of type 11000

$\mathcal{T}_X$	$\mathcal{T}_Y$	Parameter constraints		
		n	m	A_2
21100	21100	-1	-1	
21100	11111^2	-1	2	
		-1	-4	
21100	23344	-1	-4	${A_2}^4 + {A_2}^3 + 5{A_2}^2 + A_2 + 1 = 0$
11111^2	11111^2	2	2	
		2	-4	
		-4	-4	
11111^2	23344	2	-4	${A_2}^2 + 1 = 0$

Class VII: Three cancelled factors, all subtypes of 11000. Now assume that p_6 and q_6 have three factors X, Y, Z in common, including $X = F - A$ and $Y = F - B$. As above, the third factor cannot be $F + A$ or $F + B$; it must be $F - C$. Using equations (3.55) and (3.57), the expressions $(F^2 - C^2)/B^2$ and $p_6(F)$ can be expressed as polynomials in F/B with coefficients rational in n, m, and A_2. Their resultant with respect to F/B must vanish:

$$-\left(\frac{4}{({A_2}^2-1)(nA_2+m)}\right)^4 \left(\frac{nmA_2}{9}\right)^3 \prod_{l=-1,2,-4} \left({A_2}^2 + \left(\frac{m}{n} + \frac{n}{m} - \frac{mn}{l^2}\right) A_2 + 1\right) = 0.$$

Solving equations (3.55), (3.57) and

$${A_2}^2 + \left(\frac{m}{n} + \frac{n}{m} - \frac{mn}{l^2}\right) A_2 + 1 = 0$$

for l, m, n and picking a sign for C, we obtain:

$$n = \frac{(C^2-A^2)(A^2-B^2)}{A}, \quad m = \frac{(A^2-B^2)(B^2-C^2)}{B}, \quad l = \frac{(B^2-C^2)(C^2-A^2)}{C}.$$

[10] We can assume $|n| \le |m|$ because m and n switch places under the further transformation $(x, y) \mapsto (A_2x^{-1}, {A_2}^2y{-}1)$.

The class is characterized by the set $\{l, m, n\}$, so without loss of generality we may assume $|n| \le |m| \le |l|$. The cases $n = m = l$ are prohibited by $a \neq 0$, while $-2n = m = l$ is disallowed because $A_2 \neq \pm 1$. Also the case $[n, m, l] = [-1, -4, -4]$ happens to be equivalent to the class $AIR[-1/9]$. The remaining four cases are in Table 3.11.

Table 3.11: Factors X, Y, and Z, subtypes of type 11000

$\mathcal{T}_X$	$\mathcal{T}_Y$	$\mathcal{T}_Z$	Parameter constraints		
			n	m	l
21100	21100	11111^2	-1	-1	2
			-1	-1	-4
21100	11111^2	11111^2	-1	2	-4
11111^2	11111^2	11111^2	2	2	-4

Cancelled factor algorithms for subtypes of type 11000

When there is only one cancelled factor with multiplicity $[-1, 0, 0]$, or when there are two cancelled factors of the same type, we rely on step 3 of Algorithm 3.6.2, iterating through the possible types using Table 3.8. The classes in Table 3.11, as well as two of the classes in Table 3.10 are all free of parameters; though we decline to describe the corresponding algorithms here for the sake of brevity, we can say that they use the same techniques as described above, and are easily verified to work since there are no subcases to check. We therefore only describe in detail the only cases of 1-parameter classes with two cancelled factors of distinct types, both with multiplicity $[-1, 0, 0]$. These are those classes listed in Table 3.10 with $(n, m) \in \{(-1, 2), (-1, -4)\}$, both of them being of type [21100, 11111^2].

At first let us assume that there are not also cancelled factors the type of which is a subtype of 02121 or 10011, which would have multiplicities $[-1, 0, 0]$ or $[0, 1, 0]$ respectively, and at this point, that is during step 4 in Algorithm 3.6.2, would be the only other unreplaced cancelled factors.

First note that for these classes $c \neq db^3$ so that we can write $Q_{12} = {Q_6}^2$; the cancelled factors are $X = G + A_2 H$ and $Y = G + H$. The computed ICPs are:

$$L_{[-5,0,1]} = P_6 X^{-1} Y^{-2}, \qquad L_{[-1,0,0]} = X^2 Y^4, \qquad L_{[0,0,1]} = P_{12} Y^{-2},$$

$$L_{[0,1,-1]} = P_8 Y^{-2}, \qquad L_{[0,1,0]} = {Q_4}^2, \qquad L_{[3,-2,-1]} = P_{10} X^{-1} Y^{-2},$$

where $Q_4 = Q_6 X^{-1} Y^{-1}$. After verifying that $L_{[-1,0,0]}$ and $L_{[0,1,0]}$ are perfect squares, we compute

$$Z_3 := \sqrt{L_{[-1,0,0]}} \simeq XY^2,$$

$$R_{-4} := \frac{L_{[-5,0,1]}r_1}{L_{[3,-2,-1]}} = \frac{G'H - GH'}{Q_4XY},$$

$$G_4 := \frac{1}{R_{-4}}\frac{d}{dx}\ln\left(\frac{{L_{[0,1,0]}}^3}{{Z_3}^8}\right).$$

We then check that G_4 is a polynomial, and proceed with

$$G_8 := \frac{G_4}{R_{-4}}\frac{d}{dx}\ln\left(\frac{{G_4}^2}{L_{[0,1,0]}}\right).$$

We again check that G_8 is a polynomial. In general Y divides G_8 once (as does X) so that

$$Y := [G_8 : Z_3]_{[1:2]}, \quad X := Z_3Y^{-2}.$$

The sole exception occurs for $m = 2$, when ${A_2}^2+12A_2+1 = 0$; for this 0-parameter subclass we provide an alternate algorithm, but omit its description here.

When the following conditions hold, there is an extra cancelled factor of type 20011: for $m = 2$,

$$(3A_3 - 10)(-22 + 9A_3)(3{A_3}^2 - 4A_3 + 4)({A_3}^2 - 12A_3 + 44) = 0;$$

and for $m = -4$,

$$(36A_3+149)(3{A_3}^2-A_3-17)(3{A_3}^2+8A_3-56)({A_3}^4+18{A_3}^3+37{A_3}^2-1152A_3-2729) = 0;$$

where $A_3 = A_2+{A_2}^{-1}$. For these 0-parameter subclasses, we provide an alternate approach, again not described here.

The classes for which there are cancelled factors of types both a subtype of 21100 and 02121 will be discussed in the following subsection.

3.6.4 Parameter restrictions and cancelled factor replacement algorithms for types $[\mathbf{0}, \mathbf{N}, \mathbf{N-1}, \mathbf{2N-2}, \mathbf{2N-3}]$, $\mathbf{2 \le N \le 5}$

The only cancelled factor types with multiplicity $[-1, 0, 0]$ but not a subtype of type 11000 are of the form $[0, N, N-1, 2N-2, 2N-3]$. Let us now determine for which $N \in \{2, 3, 4, 5\}$ there can be such a cancelled factor. Without showing the details, we start by noting that there are 2-parameter superclasses having a cancelled factor of type 02121, and two 1-parameter superclasses with a cancelled factor of type 03243.

However, looking for further subclasses, we find first that for such a cancelled factor X to exist that $N \le 3$, and secondly that when there is a factor of type 03243 there can be no other cancelled factor of type $[0, N, N-1, 2N-2, 2N-3]$ for any N. In summary, either all cancelled factors of type $[0, N, N-1, 2N-2, 2N-3]$ have $N = 2$, or there is just one, of type 03243. If there are no cancelled factors of type $[2, k, k, 2k-2, 2k-2]$, then these are the unique cancelled factors having multiplicity $[-1, 0, 0]$ and their type is thus determined during step 3 of Algorithm 3.6.2.

It thus remains to consider classes with cancelled factors of both types $[2, k, k, 2k-2, 2k-2]$ and $[0, N, N-1, 2N-2, 2N-3]$. For classes listed in subsection 3.6.3 having cancelled factors X_i of type $[2, k, k, 2k-2, 2k-2]$, we now list in Tables 3.12 and 3.13 those subclasses for which there are also cancelled factors Y_j of type $[0, N, N-1, 2N-2, 2N-3]$.[11]

Table 3.12: Subclasses of class V with factors of type $[0, N, N-1, 2N-2, 2N-3]$

$\mathcal{T}_{X_1}$	$\mathcal{T}_{Y_1}$	$\mathcal{T}_{Y_2}$	Parameter constraints		
			n	T	W
21100	02121		-1	$\frac{5(3+P^2)}{P(P^2-5)}$	$\frac{15P^8+48P^6+360P^5+110P^4-720P^3+5400P^2-5400P+6075}{144(P^2-5)^2P^2}$
21100	03243		-1	-5	$\frac{23}{9}$
11111^2	02121		2	$\frac{3P^2-1}{4P}$	$\frac{(P+1)^3-2}{8P}$
			-4		$\frac{3072T^4+6400T^3+1200T^2-4500T-2375}{200(64T^2-45)}$
11111^2	02121	02121	2	0	$\frac{5}{12}$

Table 3.13: Subclasses of class VI with factors of type $[0, N, N-1, 2N-2, 2N-3]$

$\mathcal{T}_{X_1}$	$\mathcal{T}_{X_2}$	$\mathcal{T}_{Y_1}$	$\mathcal{T}_{Y_2}$	Parameter constraints		
				n	m	$\theta = A_2 + {A_2}^{-1}$
21100	21100	02121	02121	-1	-1	$4\theta^2 - 4\theta - 23 = 0$
21100	11111^2	02121		-1	2	$\theta^3 = -10$
				-1	-4	$\theta^2 - 7\theta + 31 = 0$
11111^2	11111^2	02121		2	-4	$\theta = 8$

For the three 1-parameter classes of type $[21100, 02121]$ or $[11111^2, 02121]$ in Table 3.12, let us assume at first that there are no cancelled factors of multiplicity $[0, 1, 0]$. By this point in Algorithm 3.6.2, all other cancelled factors have been replaced. The ICPs therefore take the following values,

$$L_{[-1,0,0]} = {X_1}^{2+2k}{Y_1}^7, \quad L_{[0,1,0]} = \frac{{Q_6}^2}{{X_1}^2}, \quad L_{[-5,0,1]} = \frac{P_6}{{X_1}^{k+1}{Y_1}^2},$$

$$L_{[3,-2,-1]} = \frac{P_{10}}{{X_1}^{k+1}Y_1}, \quad L_{[0,1,-1]} = \frac{P_8}{{X_1}^{2k}{Y_1}^2}, \quad L_{[0,0,1]} = \frac{P_{12}}{{X_1}^{2k}Y_1},$$

where $k = 0$ in the former and $k = 1$ in the latter case. Since $L_{[0,1,-1]}$ and ${L_{[-5,0,1]}}^2$ both

[11] The second and fifth classes of Table 3.12 are obtained by putting $P = 1$ and $P = 3^{-1/2}$ respectively.

have degree $6-2k$ (in G, H), their ratio is a function of F and we can compute:

$$R_{2k+9} = \frac{r_1 {L_{[0,1,0]}}^{1/2}\ L_{[-1,0,0]} L_{[-5,0,1]}}{L_{[3,-2,-1]}} = (G'H - GH'){X_1}^{2k+1}{Y_1}^6,$$

$$E_{3-6k} := D_*(L_{[0,1,-1]} {L_{[-5,0,1]}}^{-2})/R_{2k+9}$$

The denominator of E_{3-6k} is ${X_1}^{d_1}{Y_1}^{d_2}$, where $d_1 \leq 2k+1$, $5 \leq d_2 \leq 6$. Therefore we have

$$Y_1 := [L_{[-1,0,0]} : \mathrm{denom}(E_{3-6k})]_{[7:d_2]}, \quad d_2 = 5 \text{ or } 6,$$

and X follows by dividing $L_{[-1,0,0]}$ by ${Y_1}^7$.

We develop similar methods for each of the 0-parameter classes of types [21100, 03243], [11111^2, 02121, 02121], [11111^2, 11111^2, 02121], [21100, 21100, 02121, 02121], and [21100, 11111^2, 02121], as well as for those 0-p classes for which there is an extra cancelled factor Z (there can only be one) the type of which is a subtype of 20011. The corresponding parameter restrictions are as follows:

- For the first 1-parameter class in Table 3.12 ($\mathcal{T}_Z = 20011$):

$$3P^6 + 6P^4 - 77P^2 + 180 = 0,$$

$$9P^{16} + 63P^{14} + 627P^{12} + 3675P^{10} + 57481P^8 + 219933P^6$$
$$+1009665P^4 + 1622025P^2 + 1640250 = 0,$$

$$81P^{30} - 4455P^{28} + 69093P^{26} - 167211P^{24} - 1423791P^{22} + 17736921P^{20} + 49018269P^{18}$$
$$-87778515P^{16} - 646144725P^{14} + 760163075P^{12} + 6173236375P^{10} + 8184054375P^8$$
$$-14898178125P^6 - 17589403125P^4 + 40903734375P^2 + 124556484375 = 0.$$

- For the second 1-parameter class in Table 3.12 ($\mathcal{T}_Z = 20011$):

$$2P^2 + 1 = 0,$$

$$66P^8 - 217P^6 + 201P^4 - 27P^2 + 1 = 0,$$

$$35P^8 - 26P^6 + 342P^4 - 8P^2 - 1 = 0.$$

- For the third 1-parameter class in Table 3.12: In the first subclass, $\mathcal{T}_Z = 10011^2$:

$$8T^2 - 5 = 0.$$

  For the others, $\mathcal{T}_Z = 20011$:

$$645922816T^8 - 2287206400T^6 + 3006720000T^4 - 1741750000T^2 + 375390625 = 0,$$

$$927712935936T^{14} - 4485556469760T^{12} + 8902829670400T^{10} - 9259008000000T^8$$
$$+5292288000000T^6 - 1550800000000T^4 + 165937500000T^2 + 6591796875 = 0.$$

3.6.5 Parameter restrictions and cancelled factor replacement algorithms for cancelled factors which divide $\mathbf{Q_{12}}$ and $\mathbf{P_8}$ but not $\mathbf{P_6}$

Starting from Eq.(3.23), suppose Γ is a cancelled factor which divides q_6. Without loss of generality we can assume $X = G - \rho H$, $\rho \neq 0$. Applying the transformation $(x, y) \mapsto (\rho x, \rho^2(y+1)/(y-1))$, we get $X = G - H$, and the representative equation:

$$y' = \frac{Jy^2 + My + U}{(1-x^2)y + 1 + x^2}, \tag{3.59}$$

Note that the corresponding class depends only on M^2, since the above representative is preserved by the combined transformation $(x, y) \mapsto (1/x, -y)$ and reparameterization $M \to -M$. Note that the subclass $c = db^3$ corresponds to $M = 0$ or $M^2 = -J(J-3U)^2/(J-2U)$.

Now let us assume Γ also divides p_8 but not p_6. The condition becomes $p_8(1) = 0$, $p_6(1) \neq 0$. This translates into two conditions on the parameters,

$$J = 6, \quad \text{or} \quad U = \frac{(J-2)M^2}{4(J-1)^2} + \frac{J}{(J+2)(J-1)}.$$

In both cases $\mathcal{T}_X = 20011$. We name the corresponding classes X and XI respectively. In order to determine the subtypes of 20011 that X may have, we define the quotients $q_5 \equiv q_6/(F-1)$, $p_7 \equiv p_8/(F-1)$, $p_{11} \equiv p_{12}/(F-1)$, and formulate the condition that Γ further divides one of the polynomials q_5, p_7, p_{10}, p_{11}, but still does not divide p_6:

$$q_5(1)p_7(1)p_{10}(1)p_{11}(1) = 0, \quad p_6(1) \neq 0.$$

Continuing in this way, we eventually determine all possible types that X can have. We list these types in Table 3.14 along with the corresponding parameter restrictions and subclass names.[12]

X is of type 10012^2 only for the 0-parameter class XI.iii.iii, and it is the only cancelled factor for this class. The multiplicity of X is $[0, 2, 1]$ (not unique), so X is replaced during step 3 of Algorithm 3.6.2. Note that there is only one other cancelled factor type with multiplicity $[0, 2, 1]$, namely 20001. Since type 10012^2 only occurs in a 0-parameter class, it is impossible to have an ICP of multiplicity $[0, 2, 1]$ consist of cancelled factors of different types. Therefore there is no need for an algorithm for step 4 of Algorithm 3.6.2 for the non-unique multiplicity $[0, 2, 1]$.

Types 20012 and 20013 lead to multiplicities $[0, 3, 1]$ and $[0, 3, 2]$, which are unique. For cancelled factors of type 20311 and 10211, which have multiplicities $3 \times [3, -1, -1]$ and $2 \times [3, -1, -1]$ respectively, the algorithm as described so far would test the possible types for

[12]Classes XI.ii and XI.ii.i have an extra factor $X_2 = G + H$ of type 11111^2. Class XI.iii.ii has an extra factor $X_2 = G + H$ of type 10011^2. Class XI.v has an extra factor X_2 of type 20011 when $M^2 = -1/2$ ($X_2 = 3G + (4M-1)H$), or $M^2 = 9/2$, ($X_2 = 3G - (4M+9)H$).

Table 3.14: Subtypes of type 20011

<table>
<tr><th rowspan="2">Class name</th><th rowspan="2">Type of $X = G - H$</th><th colspan="3">Parameter constraints</th></tr>
<tr><th>J</th><th>M^2</th><th>U</th></tr>
<tr><td>X</td><td>20011</td><td rowspan="5">6</td><td></td><td rowspan="2"></td></tr>
<tr><td>X.i</td><td>10011^2</td><td rowspan="2">0</td></tr>
<tr><td>X.i.i</td><td>10022^2</td><td>$400U^2 - 8U - 9 = 0$</td></tr>
<tr><td>X.ii</td><td>10011^2</td><td></td><td rowspan="2">$\frac{M^2}{25} + \frac{6}{35}$</td></tr>
<tr><td>X.ii.i</td><td>20033</td><td>$-\frac{5085}{224}$</td></tr>
<tr><td>XI</td><td>20011</td><td></td><td rowspan="2"></td><td rowspan="17">$\frac{(J-2)M^2}{4(J-1)^2} + \frac{J}{(J+2)(J-1)}$</td></tr>
<tr><td>XI.i</td><td>10011^2</td><td rowspan="2">8</td></tr>
<tr><td>XI.i.i</td><td>20033</td><td>$\frac{56}{5}$</td></tr>
<tr><td>XI.ii</td><td>10011^2</td><td rowspan="2">-4</td><td></td></tr>
<tr><td>XI.ii.i</td><td>20033</td><td>10</td></tr>
<tr><td>XI.iii</td><td>10011^2</td><td></td><td rowspan="5">0</td></tr>
<tr><td>XI.iii.i</td><td>10022^2</td><td>10</td></tr>
<tr><td>XI.iii.ii</td><td>10022^2</td><td>-6</td></tr>
<tr><td>XI.iii.iii</td><td>10012^2</td><td>$5J^2 + 2J + 20 = 0$</td></tr>
<tr><td>XI.vi.i</td><td>10211^2</td><td rowspan="2">$J^2 - 2J + 4 = 0$</td></tr>
<tr><td>XI.vi</td><td>20311</td><td></td></tr>
<tr><td>XI.iv</td><td>20012</td><td rowspan="2">$5J^2 - J + 20 = 0$</td><td></td></tr>
<tr><td>XI.iv.i</td><td>20013</td><td>$-\frac{12J+10}{35}$</td></tr>
<tr><td>XI.v</td><td>40011</td><td>0</td><td></td></tr>
</table>

this multiplicity in turn. These are: 10100, 10211, and 20311. However, when $\mathcal{T}_X = 20311$, X is first assumed to be of type 10100, and an attempt will be made to replace it via the Algorithm[10100], which removes $L_{[3,-1,-1]} = X^3$ and multiplies both $L_{[0,1,0]}$ and $L_{[3,-2,-1]}$ by X^3. Subsequently it goes through Algorithm[10011], which divides $L_{[0,1,0]}$ by X and multiplies both $L_{[0,1,-1]}$ and $L_{[0,0,1]}$ by X, the net effect of which is that desired. In effect, the type is matched by accident, by way of a combination of the algorithms for two other types. Similarly, when $\mathcal{T}_X = 10211$, X is replaced by the combination of Algorithm[10100], which multiplies both $L_{[0,1,0]}$ and $L_{[3,-2,-1]}$ by X^2, and Algorithm[10011], which again divides $L_{[0,1,0]}$ by X and multiplies both $L_{[0,1,-1]}$ and $L_{[0,0,1]}$ by X.

The remaining subtypes listed in Table 3.14 have multiplicity $[0, 1, 0]$. We now consider in turn the classes for which there are one or at least two cancelled factors with multiplicity $[0, 1, 0]$. We may assume that all such cancelled factors are the only ones remaining: cancelled factors with other multiplicities have already been replaced during steps 2 to 4 of Algorithm 3.6.2.

Cancelled factor algorithms: one cancelled factor X with the multiplicity $[\mathbf{0},\mathbf{1},\mathbf{0}]$.

The 2-parameter subclasses are X and XI, in both cases $\mathcal{T}_X = 20011$. The 1-parameter subclasses are X.i, X.ii, XI.i, XI.ii, XI.iii, all of type $[10011^2]$, and XI.v, of type [40011]. The 0-parameter subclasses are X.i.i, XI.iii.i, XI.iii.ii, and XI.iii.iii ($[10022^2]$); X.ii.i, XI.i.i, and XI.ii.i ([20033]).

Cancelled factor algorithm: Class type [20011]. 2-parameter subclasses: X and XI. In this case we have the following values for the invariant polynomials:

$$L_{[0,1,0]} = X^3 {q_5}^2 H^{10}, \qquad L_{[-5,0,1]} = p_6 H^6, \qquad L_{[3,-2,-1]} = p_{10} H^{10},$$
$$L_{[0,1,-1]} = p_7 H^7, \qquad L_{[0,0,1]} = p_{11} H^{11}, \tag{3.60}$$

where $q_5 = q_6\Gamma^{-1}$, $p_7 = p_8\Gamma^{-1}$, $p_{11} = p_{12}\Gamma^{-1}$, and X divides none of q_5, p_6, p_{10}, p_7, p_{11}. We first simplify the relative invariant:

$$R_{-4} := \frac{r_1 L_{[-5,0,1]}}{L_{[3,-2,-1]}} \simeq \frac{(G'H - GH')}{X q_5 H^5}$$

and then calculate the polynomial

$$T := D_*({L_{[0,1,-1]}}^6 {L_{[-5,0,1]}}^{-7})/R_{-4}. \tag{3.61}$$

In general X divides T just once, so that we have $[L_{[0,1,0]} : T]_X = [3 : 1]$, while for any factor Y of $q_5 H^5$, we have $[L_{[0,1,0]} : T]_Y \leq [2 : 1]$. We can therefore determine X as the unique factor having the multiplicity ratio $[3 : 1]$.

If X divides T twice, we have $[L_{[0,1,0]} : T]_X = [3 : 2]$ which characterizes X uniquely unless some factor Y of q_5 divides q_6 exactly three times. A simple calculation (we start by finding those classes for which q_6 is not cubefree) shows that no such class exists.

Finally, the condition that X with $\mathcal{T}_X = 20011$ divides T at least three times, implies, for class X,

$$23241022200U = 5082M^4 + 955412665M^2 - 283631850,$$

$$15246M^6 + 76752483M^4 - 17554328350M^2 + 341915400000 = 0$$

and for class XI,

$$M^2 = \frac{4J(J-10)(J-1)(J+6)}{J^2 - 32J - 368}, \quad 27J^4 - 648J^3 - 20416J^2 - 137792J - 129536 = 0.$$

For each of these 0-parameter classes we compute

$$X^3 := [L_{[0,1,0]} : T]_{[1:1]}; \quad q_5 H^5 := [L_{[0,1,0]} : T]_{[2:1]}.$$

Cancelled factor algorithm: Class type [40011]. 1-parameter subclass XI.v. The ICPs are as in Eq.(3.60), except for $L_{[0,1,0]}$, which is instead $X^5{q_4}^2H^8$, where $q_4 = q_6\Gamma^{-2}$. Following Eq.(3.61), we find $X := [L_{[0,1,0]} : T]_{[5:2]}$.

Cancelled factor algorithms: Class type [10011²]. 1-parameter subclasses X.i, X.ii, XI.i, XI.ii, XI.iii. The invariant polynomials have the values:

$$L_{[0,1,0]} = {Q_5}^2X^4,\ L_{[-5,0,1]} = P_6,\ L_{[3,-2,-1]} = P_{10},\ L_{[0,1,-1]} = P_8X^{-2},\ L_{[0,0,1]} = P_{12}X^{-2}, \tag{3.62}$$

where $Q_5 = Q_6X^{-1}$ and X divides only $L_{[0,1,0]}$. We calculate

$$R_{-4} := \frac{r_1P_6}{P_{10}} = \frac{G'H - GH'}{Q_5X}$$

$$T := D_*(L_{[0,1,-1]}{L_{[-5,0,1]}}^{-1})/R_{-4}. \tag{3.63}$$

In general X divides T just once and Q_5 is squarefree, so we obtain $X := [L_{[0,1,0]} : T]_{[4:1]}$.

Cases where Q_5 is not squarefree: **X.ii**: $M^2 = 720/7$, $7M^2 \pm 175M + 1080 = 0$; **XI.i**: $M^2 = 896/5$, $15M^2 \pm 490M + 3976 = 0$; **XI.ii**: $M^2 = 160$, $3M^2 \pm 50M + 220 = 0$. For these 0-parameter classes we use similar methods to those previously described.

X divides T more than once for the 1-parameter classes X.i, XI.iii (we will consider these classes below), and in the following cases: **X.ii.ii**: $M^2 = \frac{10170}{91}$; **XI.i.ii**: $M = \pm 14$; **XI.ii.ii**: $M^2 = 40$.[13] For these 0-parameter classes, $[L_{[0,1,0]} : T]_X = [2 : 1]$; we use similar methods to those previously described.

Cancelled factor algorithm: Classes X.i and XI.iii. For classes X.i and XI.iii, which both have $c = db^3$, $X = G - H$ divides T twice, as does the factor $Y = G + H$ so that we have, in general, $[L_{[0,1,0]} : T]_Y = [1 : 1]$ and $[L_{[0,1,0]} : T]_Z = [2 : 1]$ for each other factor Z dividing Q_6, including X.

Let us assume at first that X and Y divide T exactly twice, and no factor of Q_6 other than X or Y divides T at least twice. Then we find $S_2 := [L_{[0,1,0]} : T]_{[1:1]} \simeq Y^2$ and $T_6 := [L_{[0,1,0]} : T]_{[2:1]} \simeq Q_4X^2$ where $Q_4 = Q_5Y^{-1}$. We define the following expressions:

$$R_4 := \frac{T_6L_{[-5,0,1]}S_2r_1}{L_{[3,-2,-1]}} = XY(G'H - GH'),$$

$$V_4 := D_*(T_6{S_2}^{-3})/R_4, \qquad V_2 := D_*(V_4{S_2}^{-2})/R_4, V_0 := D_*(V_2{S_2}^{-1})/R_4,$$
$$Z_4 := D_*(L_{[0,1,-1]}{S_2}^{-3})/R_4, \quad Z_2 := D_*(Z_4{S_2}^{-2})/R_4, Z_0 := D_*(Z_2{S_2}^{-1})/R_4; \tag{3.64}$$

[13] In this case there is not only the extra factor Y of type 11111^2 mentioned above, but also a third factor Z of type 02121 as well.

At each step we confirm that the expression V_i or Z_i is nonzero and polynomial in x; furthermore V_0 and Z_0 should be constant. We note that

$$F_1 := \frac{Z_0 V_2}{V_0 Z_2} = \frac{\frac{d}{dx}\ln(Z_2/S_2)}{\frac{d}{dx}\ln(V_2/S_2)}$$

is an absolute invariant and fractional linear in X^2/Y^2. In a similar manner we compute the absolute invariants

$$U_1 := \frac{F_1^2}{(F_1-1)^2}\frac{\frac{d}{dx}\ln(V_4/{V_2}^2)}{\frac{d}{dx}\ln(V_4/{S_2}^2)}, \quad U_2 := \frac{1}{(F_1-1)^2}\frac{\frac{d}{dx}\ln(Z_4/{Z_2}^2)}{\frac{d}{dx}\ln(Z_4/{S_2}^2)},$$

which are constants depending on the parameter W (where $W = U$ for X.i, $W = J$ for XI.iii). We can use these to solve for W and thus determine the appropriate constant $k(W)$ such that X^2 is the numerator of $F_1 - k(W)$. For class X.i we have

$$k(U) := \frac{1}{6}\frac{1014U_2 - 5309 - 4435U_1}{169U_2 - 1964 - 724U_1} = \frac{10(U+3)(40U-3)(5U-2)}{3(8U+9)(40U-3)(5U-2)}.$$

For class XI.iii, U_1 and U_2 are high degree rational expressions in J, so that we use a polynomial system solver (in this implementation, Maple's `solve` command), to solve for J, and then build $k(J)$ as a rational expression in J. For class X.i, both numerator and denominator of $k(U)$ vanish when $U = 2/5$ or $3/40$. In this case we use instead the limiting value of $k(U)$ at $U = 2/5$, namely $170/183$. For either class, if the denominator but not the numerator of $k(W)$ vanishes (X.i, $U = -9/8$ for example) we instead find X^2 as the denominator of F_1. This method fails only for Class XI.iii when $J \in \{-4, 2\}$ because Z_0 is 0; or when $J^7 - 6J^6 + J^5 + 12J^4 - 376J^3 + 32J^2 + 1552J - 960 = 0$ because Z_2/V_2 is constant. For these 0-parameter classes we develop special methods.

The remaining exceptions occur when X or Y divides T more than twice or when some factor of Q_6 other than Y divides T at least twice. Considering the former case, we find the classes X.i: $2880U^3 + 272U^2 - 148U + 3 = 0$, and XI.iii: $J = \{-18, -4, -1, 2, 4\}$, and $J^6 - 4J^5 - 23J^4 + 14J^3 + 52J^2 - 184J + 96 = 0$. $J = -1$ is one of the minimal subclasses Eq.(3.29). For the others, all free of parameters, we develop special methods.

We now consider the cases where $Q_4 = Q_6 X^{-1} Y^{-1}$ is not squarefree. We find the following cases: X.i: $U \in \{0, -6\}$, XI.iii: $J^2 + J - 1 = 0$. For these 0-parameter subclasses, we develop special methods.

Finally, we consider the cases where a factor Z of Q_4 divides T more than once. We find the following subclasses of class X.i:

- $U = 2$ (which is class $AIL[b]$, $b = -1/3$),
- $U = 3/4$ (which is $AIL[1225/39366, -13/18]$),

- $U = 6/35$,
- $252U^4 + 2299U^3 - 408U^2 + 420U - 72 = 0$;

and the following subclasses of XI.iii:

- $J = 2/3$ (also $AIL[1225/39366, -13/18]$),
- $15J^9 + 28J^8 - 215J^7 - 280J^6 + 871J^5 + 446J^4 - 2456J^3 - 608J^2 + 1920J - 576 = 0$.

For all these outstanding 0-parameter classes, we develop special methods.

Parameter restrictions and cancelled factor algorithms: At least two cancelled factors, both with multiplicity $[\mathbf{0}, \mathbf{1}, \mathbf{0}]$.

We break this case down into three subcases:

- there are just two cancelled factors, both of the simplest type, 20011;
- there are at least two cancelled factors, at least one of them a proper subtype of type 20011;
- there are at least three cancelled factors, all of them of type 20011.

Parameter constraints and cancelled factor algorithm for class type $[\mathbf{20011^2}]$. Let X, Y be the two cancelled factors. We may assume without loss of generality that $X = G - H$; and either X is of type X (ie. $J = 6$), or both X and Y are of class XI.

The first class, X.XIi, is obtained from class X with a single substitution. For XI.XIi we obtain from Eq.(3.59) via $x \mapsto Jx$ a representative equation with coefficients depending on $C_3 = -\frac{1}{2}\frac{M}{J}$. For X.Xi we use the parametric substitution $M = \frac{3(3T^2+1)}{T}$, $U = 3T^2 + 3$.

For class X.XIii, we define $M = 12e, U = 6(e^2 - f^2)$. Then we have $Q_4 = \frac{3}{2}((e+1-f)G^2 + (f-e+1)H^2)((1+e+f)G^2 + (1-e-f)H^2)$. Without loss of generality we assume that Y divides $(1+e-f)G^2 + (1-e+f)H^2$. The class type restriction implies $e \neq f - 1$, so we can reparameterize according to $e = 2(1-P^2)^{-1} + f - 1$ (thus $P^2 \neq 1$) and find that Y must now divide $G^2 - P^2H^2$. Again, without loss of generality, we choose $Y = G - PH$, and finally redefine via $f = Ph/(P^2 - 1)$.[14] The constraints become $h = -1$ (which is again X.Xi), and

$$192P^2h^2 + (3P^4 + 12P^3 + 2P^2 + 12P + 3)h - 4P(P^2 + 6P + 1) = 0. \tag{3.65}$$

[14] We have assumed $P \neq 0$ as well, since the class having $e = f + 1$ is the same (via $(x, y) \mapsto (1/x, -y)$) as that having $e = f - 1$, after renaming $f \to -f$.

Two cancelled factors $X = G - H$ and $Y \neq G + H$ (both type XI). We start with class XI which has the restriction $J \neq -2, 1$. We can further assume $M \neq 0$, $J \neq 0$ because for the corresponding subclasses XI.iii and XI.v, X is no longer of type 20011. We then substitute $M = 2J\sqrt{u/(K^2-1)}$, and $J = 3/(u+1) - 2$. Following this substitution, $Q_4 = Q_6/(G^2 - H^2)$ factors as $\Theta(K)\Theta(-K)$, where

$$\Theta(K) \equiv 3(G^2 + H^2)\sqrt{u(K^2-1)} + ((K+3)u + K)\,(G^2 - H^2).$$

Because the substitution is invariant under $K \mapsto -K$, we can assume without loss of generality that Y divides $\Theta(K)$. The constraints on the parameters for which both X and Y are of type 20011 can now be expressed as follows:

$$\text{XI.XIii: } (5K-9)(K-9)u^2 + (-59K^2 - 54K + 81)u + 8K^2 = 0, \tag{3.66}$$

$$\text{XI.XIiii: } 8(K-3)^2u^2 + (25K^2 - 48K - 9)u + 8K^2 = 0, \tag{3.67}$$

$$\text{XI.XIiv: } (K+3)(K-1)u^2 + (K+1)(5K-3)u + K^2 = 0. \tag{3.68}$$

In all three cases X and Y are the only cancelled factors.

We list in Table 3.15 the seven 1-parameter classes of type $[20011^2]$. In order to match

Table 3.15: Parameter constraints for class type $[20011^2]$

Class name		Type class of X	Parameter constraints
X.XIi	$Y = G + H$	X	$U = \frac{3}{14} + \frac{2}{49}M^2$
XI.XIi		XI	$M^2 = \frac{2J^2(1-J^2)}{J^2-4}$
X.Xi	$Y \neq G + H$	X	$M^2 = \frac{27(U-2)^2}{U-3}$
X.XIii			Eq.(3.65)
XI.XIii - XI.XIiv		XI	Eqs. (3.66), (3.67), or (3.68)

these seven classes, note that

$$L_{[0,1,0]} = {Q_4}^2 X^3 Y^3,$$

where $Q_6 = Q_4XY$, and Q_4, X, Y are pairwise coprime. We calculate T as in Eq.(3.63). In general X and Y divide T once each, and using the parameterizations listed above, we can prove that in no case can either divide T more than twice. Furthermore, for this class type no factor other than X or Y can divide $L_{[0,1,0]}$ three times. We thus have $XY = [L_{[0,1,0]} : T]_{[3:1]}[L_{[0,1,0]} : T]_{[3:2]}$.

Parameter constraints and cancelled factor algorithms: At least two cancelled factors, each a subtype of type 20011, at least one a proper subtype. First note that one 0-parameter subclass from Table 3.14, XI.iii.ii, already has an extra cancelled factor $X_2 = G + H$ of type 10011^2.

To determine the remaining subclasses, for each 1-parameter subclass in Table 3.14 we consider two cases. First, for $Y = G + H$, we look for irreducible factors of $L_{[0,1,-1]}|_{G=-H}$ which are not also factors of $L_{[-5,0,1]}|_{G=-H}$. For the case $Y \neq G + H$, we determine factors of

$$\text{resultant}(L_{[0,1,-1]}, Q_4, G)$$

which are not also factors of

$$\text{resultant}(L_{[-5,0,1]}, Q_4, G),$$

where Q_4 is $Q_6/(G^2 - H^2)$. In both cases, we define the free parameter to be a root of the irreducible factor, and substitute this value into the ICPs, and factor them to determine the type of the new cancelled factor. The resulting classes and parameter conditions are listed in Table 3.16.

None of the resulting subclasses depend on parameters. For each we devise an algorithm using invariant techniques similar to those already described; for brevity we do not describe all the details here. To prove the efficacy of each 0-parameter sub-algorithm, it is sufficient to test that it works for the generic representative obtained via the transformation $x \mapsto G(x)/H(x)$.

Note that X.i with $U = 2$ is equivalent to $AIR[-4]$, and $AIL[b]$, $b = -1/3$; and XI.iii with $J^2 + J - 5$ is equivalent to $AIR[c]$ with $81c^2 + 99c + 25 = 0$; these classes already have matching algorithms.

Parameter constraints and cancelled factor algorithms: At least three cancelled factors, all of type 20011. Finally, we consider those classes which have at least three cancelled factors of type 20011. All of these subclasses are non-parametric. In order to identify each subclass, we start with the various 1-parameter classes of type $[20011^2]$ and consider which conditions on the parameters will give rise to a third cancelled invariant of type 20011.

As an example, we describe the most difficult such case, the subclasses of classes XI.XIii-XI.XIiv of type $[20011^3]$. Let us first denote the three linear cancelled factors by X, Y, and Z. As above, we can assume $X = G - H$ and $Y, Z \neq G + H$. Furthermore, we can assume $Z \neq Y|_{H=-H}$, since if that were the case, with a scaling we would have $Y = G + H$ and $Z = G - H$. Therefore we can assume Z does not divide $\Theta(K)$. It must therefore divide $\Theta(-K)$. We must therefore satisfy both the product (3.66–3.68) and the result of replacing

Table 3.16: Parameter constraints: Two cancelled factors, subtypes of 20011, at least one of which is a proper subtype of 20011

Type	Class	Parameter constraints
$[10011^2, 20011]$	X.ii.ii	$M^2 = -105/2$
	X.ii.iii	$64288M^8 - 25697770M^6 + 1687719675M^4$ $+65424348000M^2 + 381062880000 = 0$
	XI.i.ii	$M^2 = -672/5$
	XI.i.iii	$650M^4 - 374521M^2 + 51842000 = 0$
	XI.i.iv	$450M^4 + 33985M^2 + 376712 = 0$
	XI.ii.ii	$18M^4 + 565M^2 + 50 = 0$
	XI.ii.iii	$154M^4 - 17555M^2 + 688900 = 0$
$[10011^2, 20011^2]$	X.i.x	$4900U^3 + 27633U^2 - 10602U + 972 = 0$
	XI.iii.x	$J^2 + J - 10 = 0$
	XI.iii.xi	$8J^2 + 9J - 9 = 0$
$[10011^6]$	X.iii.ii	
	X.ii.x	$98M^4 - 34335M^2 + 2764800 = 0$
	XI.i.xi	$375M^4 - 208880M^2 + 28155008 = 0$
$[10011^2, 10022^2]$	X.i.xii	$U = 3/14$
	XI.iii.ii	
$[10022^2, 20011]$	XI.v	$Y \neq G + H$, $M^2 = -1/2$, or $M^2 = 9/2$
$[10011^6, 00210]$	X.i	$U = 2$, (equivalent to $AIR[-4]$, and $AIL[b]$, $b = -1/3$)
	XI.iii	$J^2 + J - 5$, (equivalent to $AIR[c]$, $81c^2 + 99c + 25 = 0$)

K by $-K$ in this product. Computing the resultant of the two leads to the following conditions on u:

$$u \in \left\{-1, -\frac{1}{4}, 0, \frac{1}{8}, \frac{1}{2}\right\}, \qquad u^2 + 5u + 1 = 0, \qquad 4u^2 + 2u + 25 = 0,$$

$$8u^2 + 25u + 8 = 0, \qquad 16u^2 + 122u + 25 = 0, \qquad 5u^2 - 59u + 8 = 0,$$

$$280u^3 + 5484u^2 + 6402u + 1225 = 0, \quad \text{or} \quad 512u^4 + 248u^3 + 291u^2 + 77u + 8 = 0.$$

Only two of these conditions lead to subclasses of type $[20011^3]$, namely: $4u^2+2u+25 = 0$, $K = \pm(16u+25)/15$; and $512u^4+248u^3+291u^2+77u+8 = 0$, $K = \pm(12800u^3+28728u^2+6411u+3767)/5022$.

We list the complete set of subclasses and corresponding parameter constraints in Table 3.17. The corresponding cancelled factor algorithms are various, but follow the same general pattern as mentioned above, we will omit their description. In particular, the methods can be verified easily because there are no special cases to consider.

Table 3.17: Parameter constraints: At least 3 distinct cancelled factors of type 20011

Type	Class	Parameter constraints
$[20011^3]$	X.XIi.i	$64M^4 + 3122M^2 + 11025 = 0$
	XI.XIi.i	$128{C_3}^4 + 302{C_3}^2 - 25 = 0$
	X.XIii.i	$9h^2 - 5h + 1 = 0, \quad P^2 + (24h - 10)P + 1 = 0$
	X.XIii.ii	$216h^4 + 114h^3 + 33h^2 + 8h + 1 = 0,$ $P^2 + 11P + 47Ph + 186Ph^2 + 216Ph^3 + 1 = 0$
	XI.XIii.i	$4u^2 + 2u + 25 = 0$, $K = \pm(16u + 25)/15$
	XI.XIii.ii	$512u^4 + 248u^3 + 291u^2 + 77u + 8 = 0,$ $K = \pm(12800u^3 + 28728u^2 + 6411u + 3767)/5022$
$[20011^4]$	X.XIi.ii	$40M^4 - 11046M^2 + 826875 = 0$
	XI.XIi.ii	$\frac{M^2}{J^2} = -25/8$

3.7 Implementation

The above described algorithm was implemented in the computer algebra system Maple. The first prototype, designed to match AIR classes was written in 2002, and was incorporated as part of the official ODE solver libraries (under the envelope of `dsolve`) in 2006. An extension to match AIL classes as well was added in 2007.

This algorithm is capable of solving differential equations of the Abel Inverse Linear and Inverse Riccati classes, as described in the preceding sections of this chapter. The algorithm is at present restricted to the case where the coefficients of the input equation are rational in the independent variable. However, with a small change it could be made to solve most equations with arbitrary function coefficients, the only exceptions being when there are distinct functions which are related algebraically (for example, trigonometric functions). For this technique to work in general, all functions appearing in the coefficients must be expressed in terms of an algebraically independent basis.

This algorithm forms a part of Maple's ordinary differential equation solver, `dsolve`. The main linking function is called `` `odsolve/Abel/AIR` ``; with 26 subroutines, the code totals approximately 3000 lines. However, for reasons of corporate security, we do not publish the code verbatim here.

We set the Maple environment variable `_Env_try_AIR_first` to `true` before attempting the solution, in order to run the new algorithm before other methods for Abel equations.

3.7.1 Efficiency: Comparison with standard method

We ran the classification algorithm on those 0-parameter and 1-parameter classes for which the existing solving algorithms in Maple were specially designed. These were obtained from the following macro:

```
> `odsolve/Abel/seeds`;

  [45, 310, 36, 301, 1000, 42, 185, 33, 205, 147, 581, 200, 257, 400,

        515, 1001, 201, 815]

> `odsolve/Abel/p_seeds`;

            [1.1, 1.2, 1.3, 1.4, 1.6, 1.8, 1.9, 1.51, 1.5]
```

We have omitted from the tests the four classes numbered 33, 147, 815 and 1.51. These classes are not matched by the new algorithm because they are subclasses of neither the AIL nor AIR class, but they do happen to be solvable because they can be classified as AIA (Abel inverse-Abel).

To obtain each target equation, we use the macro `` `odeadv/Abel/class` ``:

```
> `odeadv/Abel/class`[45][0](x,y);
```

$$y' = (-2x^3 + 2x)y^3 + (-6x^2 + 3)y^2 - 6yx - 2$$

The input equation we have used begins with the given target equation and transforms it using $x \mapsto x + \frac{1}{x}$. These tests were run on a Pentium 1.5 GHz computer with 1GB RAM, running Maple14 with the latest version of the new algorithm.

The important conclusion to draw from these tables is that the timings for the new method are in the same order of magnitude for the 0-parameter and 1-parameter classes, whereas for the standard method it takes considerably longer to match the 1-parameter classes.

3.7.2 Examples

Example 3.7.1. We now present an example of the output of the `dsolve` command in Maple when given an Abel ODE of the form which can be solved by the algorithm presented in this work. However, because the solutions of equations in the AIR class tend to be quite large, we choose an example in a form which makes the solutions size conveniently small:

$$y' = -\frac{(y-\gamma^2)(y-\beta^2)(y-\delta^2)}{y-x^2}.$$

We then apply the simple transformation $x \mapsto x + 5/x$ and substitute the following simple values for the parameters:

$$\gamma = 2, \quad \beta = \sqrt{-1}, \quad \delta = 1,$$

Table 3.18: Timing comparison for 0-parameter dedicated AIL and AIR classes

Class name	Time for classification (Maple seconds)**	
	standard method	ICP method
45	.37	.12
310	.10	.12
36	.11	.11
301	.13	.27
1000	.09	.19
42	.10	.09
185	.11	.16
205	.18	.22
581	.19	.15
200	.14	.12
257	.22	.47
400	.20	.53
515	.27	.75
1001	.37	.49
201	.35	.36
Average	.20	.28

obtaining

$$y' = -\frac{(x^2-5)(y-4)(y+1)(y-1)}{x^2 y - (x^2+5)^2}.$$

We now show the output of a Maple session calling the command `dsolve`:

```
> PDEtools[declare](y(x), prime=x);

                    y(x) will now be displayed as y

       derivatives with respect to x of functions of one variable
                      will now be displayed with '

> _Env_try_AIR_first:= true:

> infolevel[dsolve] := 5:

> interface(imaginaryunit=i);              # display sqrt(-1) as i

> Example1 := diff(y(x),x) = -(y(x)-4)*(y(x)+1)*(y(x)-1)*(x^2-5)
             /(y(x)*x^2-(x^2+5)^2):
```

Table 3.19: Timing comparison for 1-parameter dedicated AIL and AIR classes

Class name	Time for classification (Maple seconds)**	
	standard method	ICP method
1.1	1.44	.23
1.2	1.03	.31
1.3	1.05	.35
1.4	1.92	.30
1.6	1.87	.16
1.8	2.36	.31
1.9	3.86	.37
1.5	8.15	1.37
Average	2.71	.42

```
> start_time := time():

> ans1 := dsolve(Example1, [Abel]);
Classification methods on request
Methods to be used are: [Abel]
----------------------------
* Tackling ODE using method: Abel
--- Trying classification methods ---
trying Abel
...checking Abel class AIL 2-parameter (reducible to linear)
                   and AIR 3-parameter (reducible to Riccati)
The first absolute invariant L1 is:
```

$$\frac{729}{4}\big(418x^{20} - 1757x^{19} + 25243x^{18} - 73013x^{17} + 626774x^{16} - 1384230x^{15} + 8825294x^{14} \\ -15623698x^{13} + 79401258x^{12} - 115341927x^{11} + 480508313x^{10} - 576709635x^{9} \\ +1985031450x^{8} - 1952962250x^{7} + 5515808750x^{6} - 4325718750x^{5} + 9793343750x^{4} \\ -5704140625x^{3} + 9860546875x^{2} - 3431640625x + 4082031250\big)^{3} \\ /\big(296875 - 178125x + 519375x^{2} + 292650x^{4} - 165125x^{3} - 64185x^{5} - 12837x^{7} \\ -1321x^{9} - 57x^{11} + 831x^{10} + 11706x^{8} + 80439x^{6} + 19x^{12}\big)^{5}$$

```
The second absolute invariant L2 is:
```

$$-27\left(703x^{14}-30191x^{13}+181508x^{12}-1265765x^{11}+4398595x^{10}-18093222x^{9}\right.$$
$$+43800439x^{8}-125187957x^{7}+219002195x^{6}-452330550x^{5}+549824375x^{4}$$
$$\left.-791103125x^{3}+567212500x^{2}-471734375x+54921875\right)$$
$$(x^2-x+5)^2(x^2+x+5)^3(x^4+11x^2+25)^2(x^2-2x+5)^2(x^2+2x+5)^2$$
$$/\left(418x^{20}-1757x^{19}+25243x^{18}-73013x^{17}+626774x^{16}-1384230x^{15}\right.$$
$$+8825294x^{14}-15623698x^{13}+79401258x^{12}-115341927x^{11}+480508313x^{10}$$
$$-576709635x^{9}+1985031450x^{8}-1952962250x^{7}+5515808750x^{6}-4325718750x^{5}$$
$$\left.+9793343750x^{4}-5704140625x^{3}+9860546875x^{2}-3431640625x+4082031250\right)^2$$

`The third absolute invariant L3 is:`

$$\frac{2}{3}\left(296875-178125x+519375x^2+292650x^4-165125x^3-64185x^5-12837x^7\right.$$
$$\left.-1321x^9-57x^{11}+831x^{10}+11706x^8+80439x^6+19x^{12}\right)$$
$$\left(111017x^{22}-3303761x^{21}+22557890x^{20}-195803157x^{19}+877755510x^{18}\right.$$
$$-4859506549x^{17}+16586874441x^{16}-68678532560x^{15}+188263713513x^{14}$$
$$-620057238185x^{13}+1395539305794x^{12}-3757416637253x^{11}+6977696528970x^{10}$$
$$-15501430954625x^{9}+23532964189125x^{8}-42924082850000x^{7}+51833982628125x^{6}$$
$$-75929789828125x^{5}+68574649218750x^{4}-76485608203125x^{3}+44058378906250x^{2}$$
$$\left.-32263291015625x+5420751953125\right)$$
$$/\left(418x^{20}-1757x^{19}+25243x^{18}-73013x^{17}+626774x^{16}-1384230x^{15}\right.$$
$$+8825294x^{14}-15623698x^{13}+79401258x^{12}-115341927x^{11}+480508313x^{10}$$
$$-576709635x^{9}+1985031450x^{8}-1952962250x^{7}+5515808750x^{6}-4325718750x^{5}$$
$$\left.+9793343750x^{4}-5704140625x^{3}+9860546875x^{2}-3431640625x+4082031250\right)$$
$$/\left(703x^{14}-30191x^{13}+181508x^{12}-1265765x^{11}+4398595x^{10}\right.$$
$$-18093222x^{9}+43800439x^{8}-125187957x^{7}+219002195x^{6}-452330550x^{5}$$
$$\left.+549824375x^{4}-791103125x^{3}+567212500x^{2}-471734375x+54921875\right)$$

```
The invariant signatures are:
[[-5, 0, 1], [0, 0, 1], [0, 1, -1], [0, 1, 0], [3, -2, -1]].
...checking Abel class AIL[a, b] or AIL[c]
...checking Abel class AIR[d, b, c]
...checking Abel classes 10011; 20011; 10311
...checking Abel class AIR[d, b, c], n=0
...checking Abel class AIR[d, b, c], general d, b, c
...found a candidate match; verifying
<- Abel class AIR[d, b, c], general d, b, c successful
<- Abel classes 10011; 20011; 10311 successful
The parameter values are: d = -4, b = 4, c = -1.
The equivalence function is:
```

$$F = \frac{x}{x^2+5}$$

```
The equivalent Abel equation matched is:
```

$$y' = -\frac{-4y^3+y^2+4y-1}{y-x^2}$$

```
The transformation yielding the equation in AIL or AIR form has inverse:
```

$$\left\{x = -\frac{1}{3(\frac{t}{t^2+5}-i)} + \frac{1}{6} + \frac{1}{6}i, y = \frac{5}{2}\frac{-1+u}{u-4}\right\}$$

```
<- Abel class AIR 3-parameter (reducible to Riccati) successful
<- Abel successful
```

$$\begin{aligned} ans1 := C_1 - 8\Bigg(& (1+i)(x^2-x+5)(y-4)\ {}_2F_1\left(\left[\frac{3}{10}-\frac{1}{10}i, \frac{1}{30}-\frac{1}{10}i\right], \left[\frac{4}{3}\right], \frac{5y-5}{2y-8}\right) \\ & +\frac{3}{8}(5+xi+x^2)(y-1)\ {}_2F_1\left(\left[\frac{31}{30}-\frac{1}{10}i, \frac{13}{10}-\frac{1}{10}i\right], \left[\frac{7}{3}\right], \frac{5y-5}{2y-8}\right)\Bigg)\left(\frac{y-1}{y-4}\right)^{\frac{1}{3}} \\ & /\Bigg((-4+4i)(x^2+x+5)(y-4)\ {}_2F_1\left(\left[-\frac{3}{10}-\frac{1}{10}i, -\frac{1}{30}-\frac{1}{10}i\right], \left[\frac{2}{3}\right], \frac{5y-5}{2y-8}\right) \\ & +3(5+xi+x^2)(y-1)\ {}_2F_1\left(\left[\frac{7}{10}-\frac{1}{10}i, \frac{29}{30}-\frac{1}{10}i\right], \left[\frac{5}{3}\right], \frac{5y-5}{2y-8}\right)\Bigg) = 0 \end{aligned}$$

```
> end_time := time():

> time_used_in_seconds := end_time - start_time;

                    time_used_in_seconds := 1.122
```

```
> odetest(ans1, Example1);  # verify that the solution satisfies the ODE

                                   0
```

Comparison with standard method

For comparison, we demonstrate an attempt to match the same ODE to a solvable class using the standard method. This involves computing the target invariants $J_{t,1}, \ldots, J_{t,5}$ and corresponding input invariants $J_{i,1}, \ldots, J_{i,5}$ using equations (3.10, 3.11). We can remove one variable from the problem by evaluating the input invariants at $x = 0$. The solution we are looking for then will only give us the value of F at $x = 0$, namely $F = 0$, but the values of the parameters will be sufficient to simplify the problem further at that point.

We therefore set up the system

$$\{J_{t,n} = J_{i,n}|_{x=0}\}, n = 1 \ldots 5$$

where the left hand sides, the target invariants, depend on the equivalence function F, and the three invariants b, c, d, and the right hand sides are constant. We then subtract the right hand sides from the left hand sides and take numerators, obtaining five polynomials P_n. To make sure there is no mistake, we verify that the desired solution

$$\{c = -1, d = -4, b = 4, F = 0\}$$

indeed satisfies each polynomial P_n.

We now try three different solution methods in Maple, giving a timelimit of one hour:

1. *The resultant and GCD method:*
   - Take resultants with respect to x between four pairs of polynomials P_n, including each polynomial from P_1 to P_5, obtaining four new polynomials Q_n, $n = 1 \ldots 4$ depending on b, c, d.
   - Repeat using the Q_n, obtaining three polynomials R_n, $n = 1 \ldots 3$ depending on b, c.
   - Repeat using the R_n, obtaining two polynomials S_1, S_2 depending on b.
   - Compute the GCD of S_1 and S_2 to obtain candidates for b. Use these to simplify the solving steps, and repeat the above, eventually obtaining values for b, c, d. Use these along with the original invariants — before applying the substitution $x = 0$ — to find the value of $F(x)$.

   An attempt to perform this calculation with Maple was abandoned when the first resultant calculation in the first step above had not finished after more than an hour.

2. *Use Maple's* `solve` *command:*

```
> _MaxSols := 1:
> timelimit(3600, solve({P1,P2,P3,P4,P5},{b,c,d,x})):
```

   After more than 45 minutes of calculating, this attempt ended with a "Bus error".

3. *Calculate a Gröbner basis using Maple:*

```
> timelimit(3600, Groebner[Basis]([P1,P2,P3,P4,P5],tdeg(b,c,d,x))):
```

   This attempt was also abandoned after an hour of computation.

Example 3.7.2. A second example, showing the solution of a simple 1-parameter input ODE:

```
> Example2 := diff(y(x),x) = 2*(x^4+k^2)*x*y(x)^3
  +2/3*(3*k^2-x^4-1-4*x^2)*x/(x^4+k^2)*y(x)
  -2/27*(3*x^4-9*k^2+6*x^2+2*x^6+18*k^2*x^2-2)*x/(x^4+k^2)^2;
```

$$\begin{aligned} Example2 := y' = 2(x^4+k^2)xy^3 + \frac{2}{3}(3k^2-x^4-1-4x^2)x/(x^4+k^2)y \\ - \frac{2}{27}\frac{x(3x^4-9k^2+6x^2+2x^6+18k^2x^2-2)}{(x^4+k^2)^2}. \end{aligned}$$

```
> ans2 := dsolve(Example2, [Abel]);
Classification methods on request
Methods to be used are: [Abel]
----------------------------
* Tackling ODE using method: Abel
--- Trying classification methods ---
trying Abel
...checking Abel class AIL 2-parameter (reducible to linear)
                   and AIR 3-parameter (reducible to Riccati)
The first absolute invariant L1 is:
```

$$729(8x^6+24k^2x^2-5x^4+5x^8-30x^4k^2+10x^2+3k^2+45k^4+12x^6k^2-54x^2k^4+2x^{10}-2)^3 \\ /(3x^4-9k^2+6x^2+2x^6+18k^2x^2-2)^5$$

```
The second absolute invariant L2 is:
```

$$54(4k^2+1)(-12k^2+9k^2x^2+x^6+4x^4-2x^2-6)(x^4+k^2)^2$$
$$/(-8x^6-24k^2x^2+5x^4-5x^8+30x^4k^2-10x^2-3k^2-45k^4-12x^6k^2+54x^2k^4-2x^{10}+2)^2$$

```
The third absolute invariant L3 is:
```

$$\frac{1}{3}(3x^4-9k^2+6x^2+2x^6+18k^2x^2-2)(270x^2k^4-387k^4-60x^6k^2+276x^4k^2$$
$$-270k^2x^2-54k^2-10x^{10}-41x^8+18x^6+10x^4-28x^2+60)$$
$$/(-12k^2+9k^2x^2+x^6+4x^4-2x^2-6)(-8x^6-24k^2x^2+5x^4-5x^8+30x^4k^2$$
$$-10x^2-3k^2-45k^4-12x^6k^2+54x^2k^4-2x^{10}+2)$$

```
The invariant signatures are:
[[-5, 0, 1], [0, 0, 1], [0, 1, -1], [0, 1, 0], [3, -2, -1]].
...checking Abel class AIL[a, b] or AIL[c]
...found a candidate match; verifying
<- Abel class AIL[a, b] or AIL[c] successful
The equivalence function is: F = x^2
The parameter values are: c = k^2.
The equivalent Abel equation matched is:
```

$$y' = -\frac{(y^2+k^2)}{(y-x)}$$

```
The transformation yielding the equation in AIL or AIR form has inverse:
```

$$\left\{x = t^2, y = \frac{1}{(u(t) - \frac{1}{3}\frac{2t^2+1}{t^4+k^2}} + t^2\right\}$$

```
<- Abel class AIL 2-parameter (reducible to linear) successful
<- Abel successful
```

$ans2 :=$

$$C_1 + x^2 e^{-\left(\frac{1}{k}\arctan\left(\frac{1}{k}\left(\left(y-\frac{2x^2+1}{3(x^4+k^2)}\right)^{-1}+x^2\right)\right)\right)} + \int^{\left(y-\frac{2x^2+1}{3(x^4+k^2)}\right)^{-1}+x^2} \frac{ae^{-\frac{1}{k}\arctan\left(\frac{a}{k}\right)}}{(a^2+k^2)}da = 0$$

```
> end_time := time():

> time_used_in_seconds := end_time - start_time;

time_used_in_seconds := 0.721

> odetest(ans2, Example2);

0
```

Example 3.7.3. The third example, which shows an attempt to solve an input ODE from a class without a known solution, shows that the new method wastes little time in trying to solve an equation with a signature that is not among the signatures of known solvable classes:

The first thing we do is profile the subroutines `` `odsolve/Abel/AIR` `` and `` `odsolve/Abel/1st_kind/integrate` `` which respectively implement the new and old solving methods for Abel equations.

```
> CodeTools:-Profiling:-Profile(`odsolve/Abel/AIR`):
> CodeTools:-Profiling:-Profile(`odsolve/Abel/1st_kind/integrate`):
```

```
> Example3 := diff(y(x),x)= y(x)^3+x^(-2):
```

$$y' = y^3 + x^{-2}$$

```
> start_time := time():
> dsolve(Example3, [Abel]);
Classification methods on request
Methods to be used are: [Abel]
----------------------------
* Tackling ODE using method: Abel
--- Trying classification methods ---
trying Abel
...checking Abel class AIL 2-parameter (reducible to linear)
                   and AIR 3-parameter (reducible to Riccati)
The first absolute invariant L1 is: -8*x
The second absolute invariant L2 is: -1/2
The third absolute invariant L3 is: 3
The invariant signatures are:
[[-1, 0, 0], [1, 0, 0]].
```

At this point, the new algorithm has quickly recognized that the input equation is not in a solvable class, and given up. The amount of time taken, as we will show below, is below the threshold of Maple's timer. Maple's solving algorithms now continue with the original solving methods for Abel, which end up taking one and a half seconds:

```
The relative invariant s3 is: 1/x^2
The first absolute invariant s5^3/s3^5 is: -8*x
The second absolute invariant s3*s7/s5^2 is: 3/2
```

```
...checking Abel class AIL (45)
...checking Abel class AIL (310)
...checking Abel class AIR (36)
...checking Abel class AIL (301)
...checking Abel class AIL (1000)
...checking Abel class AIL (42)
...checking Abel class AIL (185)
...checking Abel class AIA (by Halphen)
...checking Abel class AIL (205)
...checking Abel class AIA (147)
...checking Abel class AIL (581)
...checking Abel class AIL (200)
...checking Abel class AIL (257)
...checking Abel class AIL (400)
...checking Abel class AIA (515)
...checking Abel class AIR (1001)
...checking Abel class AIA (201)
...checking Abel class AIA (815)
Looking for potential symmetries
... changing x -> 1/x, trying again
Looking for potential symmetries
The third absolute invariant s5*s7/s3^4 is: -12*x
 ->        ======================================
 ->            ...checking Abel class D (by Appell)
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 1
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {4/27} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->        ======================================
 ->            ...checking Abel class B (by Liouville)
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 1
```

```
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {1, 4, 1/4, 3202210/508429} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         ======================================
 ->             ...checking Abel class A (by Abel)
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 0
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {0, -5833774/3061021, -2451979/2066563,
 -2284411/5494857, -289166/6418951, -7/12, -1/4, 2499271/5424351,
 3236533/1149811, 3274982/4015143, 4595425/3129964, 4840513/4907981,
 5949947/2501988} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         ======================================
 ->             ...checking Abel class C (by Abel)
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 0
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {2, -6988415/5558491, -6755113/3745158,
 -6436105/5089544, -6215213/682953, -6037709/4889493, -5616979/3319618,
 -5599464/4777153, -5571089/1724981, -5513075/4125056, -5447545/2782542,
 -5088487/6490796, -5029698/2689781, -4543337/5095998, -4426046/212109,
 -4182302/2806113, -3735698/3823179, -2428627/4235740, -2362466/2441539,
 -2251196/1808725, -1957399/5591343, -1597899/5208412, -892229/2281049,
 -159094/3033175, 9/5, 78849/576833, 686309/3742564, 991873/3222064,
 1314237/4312753, 1757077/5809105, 2185382/1188375, 2616669/1222070,
 2710317/827765, 4239275/2103026, 5469423/5647196, 6102948/4332143,
 6569209/4044419, 6611777/5931962, 6717676/5393949, 6825371/1898431,
 6882471/2341771} ***
 -> Step 3: looking for a solution F depending on x
```

```
*** No solution F of x was found ***
 ->         =======================================
 ->             ...checking Abel class AIL 1.6
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 0
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {-4, 16} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         =======================================
 ->             ...checking Abel class AIL 1.8
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 1
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {0} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         =======================================
 ->             ...checking Abel class AIL 1.9
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 0
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {-2/9, -1/9} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         =======================================
 ->             ...checking Abel class AIR 1.51
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 0
*** No disqualifying factor on F was found ***
```

```
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {0, 3/4, 15/4} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
 ->         ======================================
 ->             ...checking Abel class AIR 1.5
 -> Step 1: checking for a disqualifying factor on F after evaluating x
 at a number
Trying x = 1
*** No disqualifying factor on F was found ***
 -> Step 2: calculating resultants to eliminate F, get candidates for C
*** Candidates for C are {-1, 1, -5774877/3355511, -5642327/2851269,
 -4928287/3758693, -3792215/695202, -3654485/2780231, -2563164/5939513,
 -2430089/2589895, -2224577/5799792, -2046957/5529233, -1624284/1726685,
 -1515503/5590072, -1406749/6655492, -1349411/3466929, -1180058/4659333,
 -1164114/2562821, 29927/5582938, 1603963/957824, 2036067/6897592,
 2234681/6213360, 2392951/5197407, 3247168/5508079, 3487479/4433794,
 4916295/715721, 5476898/562573} ***
 -> Step 3: looking for a solution F depending on x
*** No solution F of x was found ***
looking for a particular solution
...checking Abel class AIA 2-parameter, reducible to Riccati

> end_time := time():
> time_used := end_time-start_time;

                          time_used := 1.602
```

The total time taken is 1.6 seconds. We can determine using the command `showstat` that the proportion of time taken by the new algorithm is an insignificant proportion of the total:

```
> showstat(`odsolve/Abel/AIR`);

`odsolve/Abel/AIR` := proc(f, y, x, INFO)

                         ⋮
```

```
     |Calls Seconds  Words|
PROC |    1   0.000  38225|
```

$\vdots$

```
> showstat(`odsolve/Abel/1st_kind/integrate`);
`odsolve/Abel/1st_kind/integrate` :=
proc(f, y, x, F0, F1, F2, F3, EXPLICIT_ANS, kind, INFO)
```

$\vdots$

```
     |Calls Seconds  Words|
PROC |    1   1.662 8419448|
```

$\vdots$

Example 3.7.4. Finally, we show what happens with one example which as mentioned above is not solved by the new algorithm because it falls into one of the AIA classes but it not AIL or AIR. This example is however, solved by the other Abel solving methods in `dsolve`.

```
> CodeTools:-Profiling:-Profile(`odsolve/Abel/AIR`);
> CodeTools:-Profiling:-Profile(`odsolve/Abel/1st_kind/integrate`);
> Example4 := `odeadv/Abel/class`[147][0](x,y);
```

$$Example4 := y' = \frac{y^3}{4x^2} - y^2$$

```
> start_time := time():
> dsolve(Example4,[Abel]);
Classification methods on request
Methods to be used are: [Abel]
----------------------------
* Tackling ODE using method: Abel
--- Trying classification methods ---
trying Abel
...checking Abel class AIL 2-parameter (reducible to linear)
                   and AIR 3-parameter (reducible to Riccati)
The first absolute invariant L1 is:
```

$$\frac{729}{16}\frac{(60x^3+16x^6+27)^3}{x^3(4x^3+9)^5}$$

```
The second absolute invariant L2 is:
```

$$\frac{81(16x^3-9)}{(60x^3+16x^6+27)^2}$$

```
The third absolute invariant L3 is:
```

$$\frac{8}{3}\frac{(4x^3+9)(80x^6+21x^3-27)}{(60x^3+16x^6+27)(16x^3-9)}$$

```
The invariant signatures are:
[[-5, 0, 1], [-1, 0, 0], [0, 0, 1], [0, 1, -1], [0, 1, 0], [3, -2, -1]].
...checking Abel class AIL[b]
...found a candidate match; verifying
...found a candidate match; verifying
...checking Abel class AIR[64/125, -25/64, -125/64^2]
```

At this point the new algorithm has failed, and `dsolve` continues with the old methods:

```
The relative invariant s3 is: -1/54*(4*x^3+9)/x^3
The first absolute invariant s5^3/s3^5 is:
```

$$\frac{729}{16}\frac{(60x^3+16x^6+27)^3}{x^3(4x^3+9)^5}$$

```
The second absolute invariant s3*s7/s5^2 is:
```

$$\frac{4}{3}\frac{(4x^3+9)(80x^9+420x^6+450x^3+81)}{(60x^3+16x^6+27)^2}$$

```
...checking Abel class AIL (45)
...checking Abel class AIL (310)
...checking Abel class AIR (36)
...checking Abel class AIL (301)
...checking Abel class AIL (1000)
...checking Abel class AIL (42)
...checking Abel class AIL (185)
...checking Abel class AIA (by Halphen)
...checking Abel class AIL (205)
...checking Abel class AIA (147)
inverse of the transformation solving the problem is:
```

$$\{t=x, u(t)=y(x)\}$$

```
<- Abel successful
```

$$C_1 + \frac{\left(\left(x - \frac{1}{y}\right) \mathrm{AiryAi}(\left(x - \frac{1}{y}\right)^2 - \frac{1}{2x}) + \mathrm{AiryAi}(1, \left(x - \frac{1}{y}\right)^2 - \frac{1}{2x})\right)}{\left(\left(x - \frac{1}{y}\right) \mathrm{AiryBi}(\left(x - \frac{1}{y}\right)^2 - \frac{1}{2x}) + \mathrm{AiryBi}(1, \left(x - \frac{1}{y}\right)^2 - \frac{1}{2x})\right)} = 0$$

```
> end_time := time():
> time_used := end_time-start_time;
```

```
time_used := 0.330
```

Chapter 4

Conclusions

In this work we introduced a new method of solving the invariant based equivalence problem for differential equations. This method involves a decomposition of the sequence of rational function invariants into their component polynomial factors and the subsequent use of ICP algorithms to determine the equivalence function and parameters. Together the characteristic signature that defines this ICP decomposition and the checking of intermediate necessary conditions mean that very little time is spent trying to match equations to a target equation of the wrong class. The ICP equivalence algorithms are formulaic in that they are tailored to the specific class of the target equation, making them significantly more efficient than the generic approach of the standard method. Whereas the complexity of the standard method grows exponentially with the number of parameters, the time complexity of the ICP method was shown to be essentially independent of the number of parameters of the class being matched. In fact, the most significant computation in terms of expression size and time consumed is the determination of the invariants of the input equation. Moreover, only three such invariants are required, whereas the standard method would for example require five invariants to solve an equivalence problem involving three parameters. The concept of minimal invariants, equations and subclasses provides the necessary framework so that the resulting algorithms match, and in the case of solvable classes solve, all rational coefficient equations in the given class. The one complicated feature of this approach is that the ICP structure of some parameterized classes may change when the parameters satisfy certain relations. To account for this, we described a general technique for determining the required sub-algorithms for these special subclasses.

The new technique was applied to the equivalence problem for Abel's differential equation. In particular an algorithm was implemented in Maple which is able to match all rational coefficient equations of the two-parameter Abel Inverse Linear and the three-parameter Abel Inverse Riccati classes, which together comprise almost all Abel classes with known solutions, as shown in [9]. The implementation, which included the introduction of a new sequence of invariants for Abel equations, is as far as we know the only algorithm that has been demonstrated to solve the complete family of equations of these classes.

Future directions There are many prospects for future work which can be suggested. First, we expect that it would not be difficult to generalize the domain of computation to the case where the equivalence function is not rational, as mentioned in Section 3.7. This will provide an important increase in the applicability of the solver.

Further, independently of their applicability to differential equations, the ideas presented in Sections 2.2 and 2.4.1 are relevant for the problem of overdetermined, parameterized, rational functional decomposition. Dedicated routines can be developed for this problem.

The technique as described in Chapter 2 is directly applicable to any poly-y ODE families for which we can define a fundamental set of at least two invariants, which are rational functions of the independent variable. Such families include the set of third order linear ODEs [10], as well as second order linear ODEs or Riccati equations under a restricted transformation [25]. There are surely other such classes of interest.

As to Abel equations in particular, there are many prospects for future work which can be suggested. The invariant component polynomial approach can be developed for other known solvable classes which escape the separable, Bernoulli, AIL, AIR classification, such as the AIA (Abel inverse Abel) class.

The initial motivation for developing the algorithms presented here was to facilitate classification. In fact, even after the first prototypes were developed, we were still unaware of a solution for the general Abel Inverse Riccati equation. Nevertheless, it seemed important to have a tool to quickly determine whether a new equation or family of equations under consideration was a member of a previously well-studied class. These algorithms will go a long way towards that goal.

Appendix A

Auxiliary Propositions

Proposition A.0.5. *When $n = 3$, the polynomial $u_n(G^2, H^2)$ has a factor in common with some $Z \in \{Q_6, W_4\}$. Furthermore, for these classes, neither G nor H divides Z more often than it divides $u_n(G^2, H^2)$.*

Proof. The proposition is simple to verify when $b = c = 0$, the condition $n = 3$ implying $d = 5/48$ or $1/4$ for a full parameterization. Otherwise we may assume $c \neq db^3$. Since q_6, w_4, w_7, w_{10} and w_{17} are all sparse, and using $\xi = F^2$, we can define

$$q_3(\xi) \equiv q_6(F); \quad w_i(\xi) \equiv w_{2i}(F), i = 2, 5; \quad w_i(\xi) \equiv F^{-1}w_{2i+1}(F), i = 3, 8.$$

With $q_3{}' \equiv \frac{dq_3}{d\xi}$, $w_i{}' \equiv \frac{dw_i}{d\xi}$, the relations among these new polynomials are (*cf.* (3.48)):

$$w_2 \equiv \frac{3}{2}q_3 q_3{}'' - q_3'^2 - 4\xi, \tag{A.1}$$

$$w_3 \equiv 3w_2{}'q_3 - 2w_2 q_3{}', \tag{A.2}$$

$$w_5 \equiv w_3 w_2 + 2\xi(w_3{}'w_2 - 2w_3 w_2{}'), \tag{A.3}$$

$$w_8 \equiv w_5{}'q_3 w_2 - w_5 q_3{}'w_2 - w_5 q_3 w_2{}'. \tag{A.4}$$

The condition $n = 3$ means that $W_8 \mid W_{24}$, that is, $w_7 H^8 \mid q_6 w_{17} H^{24}$. This implies that $w_3 \mid q_3 w_8$ in $\mathbf{K}[\xi]$.

Let us assume for now that w_3 and q_3 do not share a common factor. Therefore $w_3 \mid w_8$. Reducing w_8 modulo w_3 using Eqs. (A.1 – A.4), we find that w_3 divides $q_3 w_2 v_3$ where

$$v_3 \equiv 2\xi w_2 w_3'' + (3w_2 - 7\xi w_2')w_3' + \frac{21}{2}\xi w_2{}''w_3,$$

the last term being added to reduce the degree of v_3 to 3. Now, $n = 3$ implies that

$\deg(w_3) = 3$,[1] but w_3 cannot be proportional to v_3.[2] We conclude that w_3 must have a common factor with either w_2 or q_3. This conclusion is also logically implied by the negation of the assumption, so it must be true whether or not the assumption is satisfied.

This proves the first statement of the proposition. Using Eq.(A.2), we see that the same factor must be common to one of the three pairs (w_2, w_2'), (q_3, q_3'), or (w_2, q_3). Checking each case shows that neither W_4 nor Q_6 is divisible by G or H when $c \neq db^3$. □

Proposition A.0.6. *When $n = 2$, W_4 is not proportional to ${E_2}^2$, where $E_2 = W_8/U$, even if n was 3 before redefining U.*

Proof.

Suppose E_2 divides W_4 twice and W_8 N times. Assuming neither G nor H divide E_2, $e_2 \equiv E_2(F, 1)$ divides w_2 twice and w_3 N times. By Eqs.(3.48) e_2 divides w_{10} at least $N+1$ times and thus w_{17} at least $N+2$ times. Converting back, we find E_2 divides W_{24} at least $N+2$ times. Now, the only case with $a = 1$ where Q_6 is not cubefree is $d = 1/3/b, c = b^2/3$; in this case E_2 is not the factor which divides Q_6 three times. In all cases therefore, E_2 divides W_{24}/Q_6 at least N times, and therefore does not divide $gcd({W_8}^2, {W_{24}}^2/Q_{12})$ at all.

Therefore n must have been 3 originally. However the only classes for which n was originally 3 and W_4 is a square are either $AIR[-16/375, -25/16, 125/768]$ (which is the same class as $AIR[-1/9]$), $AIR[e^2(e+1)/12, 4/e^2, 16(e+1)/(3e^4)]$ (which has $c = db^3$), or $AIR[-\frac{148}{2187} - \frac{32}{2187}\sqrt{10},\ -\frac{9}{4}, \frac{37}{48} - \frac{1}{6}\sqrt{10}]$ (equivalently, $AIR\{-\frac{9}{4}, -\frac{1}{36}, \frac{37}{24}\}$), for which, however, E_2 is the factor common to W_8 and Q_6, but does not divide W_4. □

---

[1]The subclasses for which $\deg(w_3) < 3$ can be parameterized as $\mathrm{AIR}[1/(3b), b, (b^2 + 4b)/3]$ and $\mathrm{AIR}[1/e^2, (e^2 + 2e)/3, (e^4 + 6e^3 + 12e^2)/27]$. In the first case $n = 1$, while in the second $n = 1$ unless $e \in \{-9, -9/2, -27/10\}$, in which case $n = 2$.

[2]Since w_3 cannot be 0 ($w_3 = 0$ implies that ${q_3}^2/{w_2}^2$ is constant, but q_3 being a cube implies that $w_2 = 0$), we set $v_3 = kw_3$ and equate coefficients of ξ; the resulting system has no solution with $c \neq db^3$.

Bibliography

[1] Niels Henrik Abel. *Oeuvres Complètes*, volume II. Christiana, 1881.

[2] Cesar Alonso, Jaime Gutierrez, and Tomas Recio. A rational function decomposition algorithm by near-separated polynomials. *Journal of Symbolic Computation*, 19:527–544, 1995.

[3] Paul Appell. Sur les équations différentielles algébriques et homogènes par rapport à la fonction inconnue at à ses dérivées. *Comptes Rendus*, 104:1776–1779, 1887.

[4] Paul Appell. Sur les invariants des équations différentielles. *Comptes Rendus*, 105:55–58, 1887.

[5] Paul Appell. Sur les invariants de quelques équations différentielles. *Journal de mathématiques pures et appliquées*, 5:361–423, 1889.

[6] Matthias Berth and Günter Czichowski. Using invariants to solve the equivalence problem for ordinary differential equations. *Applicable Algebra in Engineering, Communications and Computing*, 11:359–376, 2001.

[7] Edgardo S. Cheb-Terrab. A connection between Abel and ${}_pF_q$ hypergeometric differential equations. *European Journal of Applied Mathematics*, 16:53–63, 2005.

[8] Edgardo S. Cheb-Terrab and Austin D. Roche. Abel equations: Equivalence and integrable classes. *Computer Physics Communications*, 130:204–231, 2000.

[9] Edgardo S. Cheb-Terrab and Austin D. Roche. An Abel ODE class generalizing known integrable classes. *European Journal of Applied Mathematics*, 14:217–229, 2003.

[10] Edgardo S. Cheb-Terrab and Austin D. Roche. Hypergeometric solutions for third order linear odes. http://arxiv.org/abs/0803.3474, 2008.

[11] George H. Halphen. Sur la réduction des équations différentielles linéaires aux formes intégrables. *Mémoires présentés par divers savants, Académie des sciences*, 28: 1:1–301, 1887.

[12] Erich Kamke. *Differentialgleichungen: Lösungsmethoden und Lösungen.* Chelsea Publishing Co, New York, 1959.

[13] Roger Liouville. Sur certaines équations différentielles du premier ordre. *Comptes Rendus*, 103:476–479, 1886.

[14] Roger Liouville. Sur quelques équations différentielles non linéaires. *Comptes Rendus*, 103:457–460, 1886.

[15] Roger Liouville. Sur une classe d'équations différentielles non linéaires. *Comptes Rendus*, 103:520–523, 1886.

[16] Roger Liouville. Sur une classe d'équations différentielles du premier ordre et sur les formations invariantes qui s'y rapportent. *Comptes Rendus*, 105:460–463, 1887.

[17] Roger Liouville. Sur une équation différentielle du premier ordre. *Acta Mathematica*, 27(1):55–78, 1903.

[18] George M. Murphy. *Ordinary Differential Equations and their solutions.* Van Nostrand, Princeton, 1960.

[19] Eugen Netto. Ueber einen Lüroth-Gordan'schen Satz. *Mathematische Annalen*, 46:310–318, 1895.

[20] Peter Olver. *Equivalence, Invariants and Symmetry.* Cambridge University Press, 1995.

[21] Andrei D. Polyanin and Valentin F. Zaitsev. *Handbook of Exact Solutions for Ordinary Differential Equations.* CRC Press, 1995.

[22] Austin D. Roche. A solution algorithm for the Abel inverse Riccati ODE. Preprint, Computer Algebra Group, Simon Fraser University, http://oldweb.cecm.sfu.ca/CAG/products.html, 2004.

[23] P. L. Sachdev. *A Compendium of Nonlinear Ordinary Differential Equations.* John Wiley & Sons, 1997.

[24] Fritz Schwarz. Algorithmic solution of Abel's equation. *Computing*, 61:39–46, 1998.

[25] Katherina von Bülow. Equivalence methods for second order linear differential equations. Master's thesis, 2000.

[26] Richard Zippel. Rational function decomposition. In *Proceedings of the 1991 International Symposium on Symbolic and Algebraic Computation*, pages 1–6. ACM, 1991.

[27] Daniel Zwillinger. *Handbook of Differential Equations.* Academic Press, 1998.