London Mathematical Society ISSN 1461–1570

COMPUTATIONAL ASPECTS OF CLASSIFYING SINGULARITIES

N. P. KIRK

Abstract

A `Maple` package which performs the symbolic algebra central to problems in local singularity theory is described. This is a generalisation of previous projects, which dealt only with problems in elementary catastrophe theory. Applications to specific problems are described, and a survey given of the powerful techniques from singularity theory that are used by the package. A description of the underlying algorithm is given, and some of the more important computational aspects discussed. The package, user manual and installation instructions are available in the appendices to this article.

1. *Introduction*

Calculations that arise in local singularity theory lend themselves naturally to symbolic algebra methods. In this article we describe a package which deals with problems in classification and unfolding theory for the standard equivalence relations encountered in singularity theory. The package, called `Transversal`, consists of a collection of procedures which run under the symbolic algebra system `Maple` [**7**].

We refer to the survey article of Wall [**25**] and the book of Martinet [**17**] for a comprehensive discussion of the singularity theory used in this article. The more recent advances in determinacy and classification theory are discussed in the articles by Bruce, Kirk, du Plessis and Wall [**4, 6**]. The techniques developed in these provide a very efficient, wide-ranging classification scheme involving algebraic calculations which may be reduced to finite-dimensional symbolic problems. However, the calculations can become very intensive and repetitive, which is where the need for a specialist computer package arises.

The applications we have in mind require the calculation of certain 'tangent spaces' in a jet-space. This calculation involves the manipulation of truncated polynomial vectors and is therefore really just a problem in linear algebra that can be handled by a computer. For example, in classification problems the calculation can be reduced to the enumeration of the orbits of the jet-group. In this situation we are considering Lie groups acting on smooth manifolds and have powerful techniques such as *Mather's lemma* [**25**, Lemma 1.1] and *complete transversals* [**4**] at our disposal. (In fact, we are dealing with algebraic groups over **R** or **C** acting algebraically on an affine space and stronger results can be established. Although of theoretical importance, we will not need such results in our present applications.) It turns out that all of the information that we require can be obtained from a calculation of the tangent spaces to the orbits of the jet-group in the jet-space. Calculations in unfolding theory can be reduced to similar symbolic manipulations. We do not have a Lie group action in this case (we only have the notion of 'extended equivalence' at the germ level) but unfolding theory allows us to work with the associated 'extended tangent

Received 13 February 1999, revised 17 November 1999*; published 25 July 2000*.
2000 Mathematics Subject Classification 58K40, 68W30

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

spaces'. Once we have concluded that a given germ is finitely determined (using the above methods) we may perform unfolding calculations in a suitable jet-space. At the jet level, the calculation of these 'extended tangent spaces' involves identical symbolic manipulations to those required in classification calculations.

The 'tangent spaces' are given by the action of a space of vector fields L on a given germ or jet. For example, if L is the Lie algebra of a jet-group then the tangent space to the orbit through the jet f is given by the natural action of the Lie algebra and is denoted by $L \cdot f$. We will use 'tangent space' as a general term to refer to such spaces $L \cdot f$ (even though they are not necessarily tangent to some submanifold). The terminology is used at both the jet and the germ levels. (This notation was established in the more recent work [4, 6] as a preferred alternative to the ad-hoc notation $T\mathcal{G} \cdot f$ used previously.) The main feature of our package is its ability to calculate and manipulate the spaces $L \cdot f$. Our aim was to produce a package capable of performing the calculations over a wide range of equivalence relations. In particular, it must apply to the cases where $\mathcal{G}$, a subgroup of $\mathcal{K}$ defining the equivalence relation, is one of the standard Mather groups $\mathcal{R}$, $\mathcal{L}$, $\mathcal{A}$, $\mathcal{C}$ or $\mathcal{K}$ [25]; or, more generally, one of Damon's geometric subgroups [9] for which a set of generators of the Lie algebra $L\mathcal{G}$ can be written down explicitly.

Let us consider one of the important research areas in singularity theory, namely the case of $\mathcal{A}$-equivalence. Not only is this a natural generalisation of $\mathcal{R}$-equivalence, but it has significant applications in geometry and related areas such as computer vision. For example, in such applications one often wants to consider the simultaneous contact between a submanifold and a whole family of model submanifolds, typically families of lines, planes, circles, spheres, and so forth. In these situations we must work with $\mathcal{A}$-equivalence rather than contact ($\mathcal{K}$) equivalence, the difference essentially being that contact between nearby fibres of the map is preserved under $\mathcal{A}$-equivalence, whereas $\mathcal{K}$-equivalence relates only to the contact class associated to one fibre. For a recent survey of geometrical applications of singularity theory we refer to the article of Bruce [3] and the extensive bibliography therein. A real obstruction in obtaining $\mathcal{A}$-classifications is the size of the computations involved in all but the simplest of examples. One only has to refer to the existing papers dealing with $\mathcal{A}$-classification to see this, for example, those of Mond, du Plessis and Rieger [19, 20, 23]. For such applications any useful package must be able to calculate $L \cdot f$ in a given jet-space for a given jet f where $L = L\mathcal{A}$ (for applications of Mather's lemma, calculation of $\mathcal{A}$-invariants, moduli detection); $L = L\mathcal{A}_1$ with the possible inclusion of a nilpotent part (for determinacy and complete transversal calculations in classification problems); and $L = L\mathcal{A}_e$ (for unfolding calculations). The package achieves all of these requirements and we cite its success in $\mathcal{A}$-classifications as its single most important application. For example, the aforementioned results of Mond and Rieger were all reproduced in a matter of hours using `Transversal`. Recent applications of `Transversal` [5, 12, 13, 14, 15, 26] represent some of the most extensive classifications carried out to date. The package has been extended to deal with weighted homogeneous filtrations, multigerms and cases where the equivalence $\mathcal{G}$ derives from a set of liftable or lowerable vector fields (the latter providing new results in the theory of caustics and envelopes). The package and all its variants are described, together with detailed examples and discussions of the underlying algorithms, in the Transversal User Manual [16].

The remainder of this article is organised as follows. In Section 2 specific applications of our package to singularity theory are described and the mathematical background reviewed. We give an overview of the package in Section 3. This begins with a brief survey of previous applications of computer techniques to singularity theory. We then describe the basic functionality of our package and the underlying algorithm. Some of the more technical aspects of the algorithm are discussed separately in the final part of this section. Finally, in Section 4 we give several examples involving $\mathcal{A}$-classification and discuss how the package deals with the calculations.

Appendix A includes a README file which gives detailed installation instructions for the package, and a comprehensive user manual [**16**]. Appendix B provides a link that allows one to download the package itself, in a version that runs under `Maple` V, Releases 1–4. Appendix C provides the link to the `Maple` V, Relase 5, version. Since the package is a (somewhat large) collection of `Maple` procedures (stored in text format) it should run under all versions of `Maple` (for example, Unix, PC, Mac). In addition, the Unix version comes with a simple shell script which installs the package as a `Maple` library (setting up paths, and so on), thus making matters easier.

2. *Applications to singularity theory*

We discuss how our package may be used to solve problems in singularity theory and, for completeness, review the mathematical techniques which are required. Section 3 will describe how one actually implements these techniques in the package.

The notation used throughout this article is standard, based on (some of) that developed in [**17, 25**]. In addition, we adopt the more systematic notation used in [**4, 6**] and clarify the following. The theory applies over both the real and complex numbers, and $\mathbf{F}$ will denote either $\mathbf{R}$ or $\mathbf{C}$. (In addition, the classifications in these cases hardly differ. Minor simplifications occur in the $\mathbf{C}$ case due to the collapsing of orbits, most commonly resulting in the removal of a $\pm$ sign in the normal form.) The local ring of differentiable/analytic function-germs $\mathbf{F}^n, 0 \to \mathbf{F}$ is denoted by $\mathcal{E}_n$ and its maximal ideal by $\mathcal{M}_n$. The corresponding module of map-germs $\mathbf{F}^n, 0 \to \mathbf{F}^p$ is denoted $\mathcal{E}(n, p)$; those with zero target are therefore given by $\mathcal{M}_n\mathcal{E}(n, p)$. We define the standard kth jet-space $J^k(n, p)$ to be $\mathcal{M}_n\mathcal{E}(n, p)/\mathcal{M}_n^{k+1}\mathcal{E}(n, p)$. This is identified with the space of p-tuples of polynomials in n indeterminates over $\mathbf{F}$ which vanish at $0 \in \mathbf{F}^n$, truncated to degree k, a germ f being identified with its Taylor expansion to degree k. Unless otherwise stated, $\mathcal{G}$ will denote a subgroup of the contact group $\mathcal{K}$, usually one of the standard Mather groups $\mathcal{R}$, $\mathcal{L}$, $\mathcal{A}$, $\mathcal{C}$ or $\mathcal{K}$, but in principle one of Damon's geometric subgroups. We let $\mathcal{G}_k$ be the normal subgroup of $\mathcal{G}$ consisting of those germs whose k-jet is equal to that of the identity. The standard kth jet-group $J^k\mathcal{G}$ is defined to be the quotient group $\mathcal{G}/\mathcal{G}_k$. This is a Lie group and acts on the affine space $J^k(n, p)$. We will abbreviate the term 'complete transversal' to 'CT' from now on.

2.1. *Classification theory: complete transversals and determinacy*

In classification theory we seek to list orbits of finitely determined germs $f \in \mathcal{M}_n\mathcal{E}(n, p)$ under the action of the group $\mathcal{G}$, choosing suitable normal forms as representatives. Classification is done inductively at the jet-level, classifying in turn all $(k + 1)$-jets with a given k-jet until determined jets result (or pre-selected upper bounds on moduli or codimension are reached). The method of 'complete transversals' provides an efficient means of carrying out this procedure. We recall some of the main results from [**4, 6**].

The group $\mathcal{G}$ is said to be *jet-closed* if for each $r \geqslant 1$, $J^r\mathcal{G}$ is a closed subgroup of $J^r\mathcal{K}$. If $\mathcal{G}$ is jet-closed it follows that $J^s(L\mathcal{G}) \subset L(J^s\mathcal{G})$ for all s. In many cases we have equality. If a jet-closed group $\mathcal{G}$ satisfies $J^s(L\mathcal{G}) = L(J^s\mathcal{G})$ for all s then we call it *fibrant*. We find that $\mathcal{R}$, $\mathcal{L}$, $\mathcal{A}$, $\mathcal{C}$ and $\mathcal{K}$ are all jet-closed and fibrant. Further examples are given via the following concept. Let $\mathcal{H}$ be a subgroup of $\mathcal{G}$; then $\mathcal{H}$ is said to be *strongly closed* in $\mathcal{G}$ if

$\mathcal{H}_s = \mathcal{G}_s$ for some s (equivalently $\mathcal{G}_s \subset \mathcal{H}$), and $J^s\mathcal{H}$ is closed in $J^s\mathcal{G}$. Now, a strongly closed subgroup $\mathcal{H}$ of a jet-closed group $\mathcal{G}$ is itself jet-closed. If, in addition, $\mathcal{G}$ is fibrant then so is $\mathcal{H}$.

The map

$$L(J^1\mathcal{K}) \cong gl(n, \mathbf{F}) \oplus gl(p, \mathbf{F}) \to gl(n+p, \mathbf{F})$$
$$(M, N) \mapsto \begin{pmatrix} M & 0 \\ 0 & N \end{pmatrix},$$

where $gl(n, \mathbf{F})$ denotes the Lie algebra of the general linear group $GL(n, \mathbf{F})$, is a faithful representation of the Lie algebra $L(J^1\mathcal{K})$ on $\mathbf{F}^{n+p}$. Suppose that $L(J^1\mathcal{G})$ acts nilpotently on $\mathbf{F}^{n+p}$ under this representation. This happens if the source and target parts of $L(J^1\mathcal{G})$ are spanned by *strictly* upper (or lower) triangular matrices, for example. Generally the requirement is equivalent to $J^1\mathcal{G}$ being unipotent. In this situation the following sum is finite (see [6]) and we may define the *nilpotent filtration* of $\mathcal{M}_n\mathcal{E}(n, p)$,

$$M_{r,s}(\mathcal{G}) = \sum_{i \geqslant s} (L\mathcal{G})^i \cdot (\mathcal{M}_n^r\mathcal{E}(n, p)) + \mathcal{M}_n^{r+1}\mathcal{E}(n, p),$$

for integers $r \geqslant 1$ and $s \geqslant 0$. Observe that this is finer than the standard filtration by degree. For $r = 0$ we define $M_{0,0}(\mathcal{G})$ to be $\mathcal{M}_n\mathcal{E}(n, p)$ for consistency. The associated (r, s)-*jet-space* $J^{r,s}(n, p)$ is then defined to be $\mathcal{M}_n\mathcal{E}(n, p)/M_{r,s}(\mathcal{G})$. This is a refinement of the standard jet-space $J^r(n, p) = \mathcal{M}_n\mathcal{E}(n, p)/\mathcal{M}_n^{r+1}\mathcal{E}(n, p)$. Thus, $J^{r,0}(n, p)$ is $J^{r-1}(n, p)$, and as s increases $J^{r,s}(n, p)$ contains more of the homogeneous terms of degree r, until for some finite $s = k_r$ where we find that $J^{r,k_r}(n, p)$ is the whole of $J^r(n, p)$ (k_r exists due to nilpotency). Provided that we work with these refined jet-spaces, we have the following complete transversal result.

Theorem 2.1. [4, Theorem 2.9] *Let $\mathcal{G}$ be a fibrant subgroup of $\mathcal{K}$ such that $L(J^1\mathcal{G})$ acts nilpotently on $\mathbf{F}^{n+p}$. Let f be a smooth germ $\mathbf{F}^n, 0 \to \mathbf{F}^p, 0$ and let T be a subspace of $M_{r,s}(\mathcal{G})$ with*

$$M_{r,s}(\mathcal{G}) \subset T + L\mathcal{G} \cdot f + M_{r,s+1}(\mathcal{G}).$$

Then any germ $g : \mathbf{F}^n, 0 \to \mathbf{F}^p, 0$ with $g - f \in M_{r,s}(\mathcal{G})$ is $\mathcal{G}$-equivalent to a germ of the form $f + t + \phi$ with $t \in T$ and $\phi \in M_{r,s+1}(\mathcal{G})$.

This is really just a question in the standard jet-space $J^r(n, p)$, provided that we order the homogeneous terms of degree r as dictated by $M_{r,s}(\mathcal{G})$. The latter can be achieved by using a system of weights, see Section 3.3. The spaces T and $f + T$ are both referred to as a *complete transversal* (CT). One of the main features of the package is to calculate a basis for T, taking $L = J^r(L\mathcal{G})$. In practice, this is a process which has to be carried out numerous times and, as the classification proceeds, soon becomes computationally infeasible without the help of a computer.

Example 2.2. An example should clarify the discussion above. Consider the classification of map-germs $\mathbf{F}^2, 0 \to \mathbf{F}^2, 0$ under $\mathcal{A}$-equivalence. Let (x, y) denote coordinates in the source, and (u, v) those in the target. Recall that $\mathcal{A}_1$ denotes the subgroup of $\mathcal{A}$ consisting of those germs whose 1-jet is the identity, and define $\mathcal{G}$ to be the unipotent subgroup of $\mathcal{A}$ having nilpotent Lie algebra

$$L = L\mathcal{A}_1 \oplus \mathbf{F}\{x\partial/\partial y\} \oplus \mathbf{F}\{v\partial/\partial u\}.$$

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

Table 1: Generators for the homogeneous part of $J^{r,s}(\mathcal{G})$.

(r, s)	Homogeneous part	Weight
$(1, 0)$	$\{0\}$	
$(1, 1)$	$\{(0, y)\}$	1
$(1, 2)$	$\{(y, 0), (0, x)\}$	2
$(1, 3)$ or $(2, 0)$	$\{(x, 0)\}$	3
$(2, 1)$	$\{(0, y^2)\}$	2
$(2, 2)$	$\{(y^2, 0), (0, xy)\}$	3
$(2, 3)$	$\{(xy, 0), (0, x^2)\}$	4
$(2, 4)$ or $(3, 0)$	$\{(x^2, 0)\}$	5

This acts on a germ $f = (f_1, f_2)$ by

$$L \cdot f = \mathcal{M}_2^2\langle \partial f/\partial x, \partial f/\partial y\rangle + f^*(\mathcal{M}_2^2)\{e_1, e_2\} + \mathbf{F}\{x\partial f/\partial y, f_2 e_1\},$$

where e_1 and e_2 are the canonical basis vectors in $\mathbf{F}^2$. Each (r, s)-jet-space is just a refinement of the standard r-jet-space and a convenient way to describe these spaces is to list the 'homogeneous' generators for each of the spaces $J^{r,s}(2, 2)$; see Table 1. The 'weight' column demonstrates the use of weights to partition the monomial vectors into their (r, s)-levels as described in Section 3.3; here $\alpha = (2, 1)$ and $\beta = (-1, 0)$. This example is discussed further in Section 4.

Example 2.3. The above results incorporate the notion of *strong equivalence*. For example, two germs are defined to be *strongly $\mathcal{A}$-equivalent* if they are $\mathcal{A}_1$-equivalent; that is, the diffeomorphism defining the equivalence has linear part the identity. Here we can take $\mathcal{G}$ to be the unipotent group $\mathcal{A}_1$. Thus, $M_{r,s}(\mathcal{G}) = \mathcal{M}_n^{r+1}\mathcal{E}(n, p)$ for all $s > 0$ and the CT theorem applies to the standard jet-spaces $J^r(n, p)$. Given a germ $f : \mathbf{F}^n, 0 \to \mathbf{F}^p, 0$, suppose that T is a vector subspace of the space of homogeneous jets of degree $k+1$ such that

$$\mathcal{M}_n^{k+1}\mathcal{E}(n, p) \subset T + L\mathcal{A}_1 \cdot f + \mathcal{M}_n^{k+2}\mathcal{E}(n, p).$$

Then every germ F with $F - f \in \mathcal{M}_n^{k+1}\mathcal{E}(n, p)$ is $\mathcal{A}_1$-equivalent to a germ of the form $f + t + \phi$ with $t \in T$ and $\phi \in \mathcal{M}_n^{k+2}\mathcal{E}(n, p)$. That is, if $j^k F = j^k f$ then $j^{k+1}F$ is $J^{k+1}\mathcal{A}_1$-equivalent to a jet of the form $j^{k+1}f + t$, for some $t \in T$. This provides an $\mathcal{A}$-classification procedure with respect to familiar polynomial degree. However, in many classifications we need to use larger unipotent subgroups than $\mathcal{A}_1$ to obtain an efficient $\mathcal{A}$-classification procedure, at least during the early stages of the classification. We therefore have to classify in finer steps, using the refined jet-spaces $J^{r,s}(n, p)$, as in Example 2.2.

We now turn to the determinacy question. Algebraic criteria which characterise determinacy were found in [6]. These results also provide excellent determinacy estimates for use in practical situations. A version of the results suited to our needs is as follows. We shall restrict ourselves here to the case where $\mathcal{G}$ is one of the standard Mather groups, to avoid the extra technicalities, though the determinacy results do apply to a larger class of groups.

Theorem 2.4. [6, Theorem 2.1] *Let $\mathcal{G}$ be one of $\mathcal{R}$, $\mathcal{L}$, $\mathcal{A}$, $\mathcal{C}$ or $\mathcal{K}$ and let $\mathcal{H}$ be a strongly closed subgroup of $\mathcal{G}$ such that $L(J^1\mathcal{H})$ acts nilpotently on $\mathbf{F}^{n+p}$. Then a smooth germ $f : \mathbf{F}^n, 0 \to \mathbf{F}^p, 0$ is k-$\mathcal{H}$-determined if and only if*

$$\mathcal{M}_n^{k+1}\mathcal{E}(n, p) \subset L\mathcal{H} \cdot f.$$

Although the results are stated in terms of germs, they may be reduced to questions involving jets. We will show that establishing the degree of determinacy of a germ is a special case of calculating CTs. When the tangent space $L \cdot f$ is a module over $\mathcal{E}_n$ (for example, when $\mathcal{G} = \mathcal{R}$, $\mathcal{C}$ or $\mathcal{K}$) this follows from a simple application of *Nakayama's lemma*; see [17, p. 131] and [25, p. 489]. We find that the germ is k-$\mathcal{G}$-determined if, when considered as a k-jet, the CT of degree $k+1$ is empty. For the remaining cases of interest, where $L \cdot f$ is a module over $\mathcal{E}_p$ via f^*, we apply a result of du Plessis [6, Lemma 2.6]. Probably the most important and informative application is where $\mathcal{G} = \mathcal{A}$, so we take this as an example. Applying du Plessis' result to the above determinacy theorem gives the following theorem.

Theorem 2.5. *Using the notation of Theorem 2.4, f is k-$\mathcal{H}$-determined if and only if*

$$\mathcal{M}_n^{k+1}\mathcal{E}(n, p) \subset L\mathcal{H} \cdot f + \mathcal{M}_n^{k+1} f^*(\mathcal{M}_p)\mathcal{E}(n, p) + \mathcal{M}_n^{2k+2}\mathcal{E}(n, p).$$

Thus, f is k-$\mathcal{A}$-determined if the successive transversals from degree $k+1$ to degree $2k+1$ are empty. (Of course, the terms in $\mathcal{M}_n^{k+1} f^*(\mathcal{M}_p)\mathcal{E}(n, p)$ can be used to reduce the upper limit from $2k+1$. This is extremely important in applications, but the revised upper limit that one obtains depends on the particular germ f.)

The spaces $L \cdot f$ used in determinacy calculations are precisely those used in CT calculations. We therefore obtain a very efficient classification process: if the determinacy criterion fails due to a non-empty transversal we simply continue the classification, the transversal providing us with a list of (possible) new branches in the classification tree.

Example 2.6. We reconsider Examples 2.2 and 2.3. For the former we take $\mathcal{H}$ in Theorems 2.4 and 2.5 to be the unipotent group $\mathcal{G}$ defined in Example 2.2. For strong determinacy considered in Example 2.3 we take $\mathcal{H}$ to be $\mathcal{A}_1$. As a further example, consider $\mathcal{R}$-determinacy of function-germs. The condition for strong determinacy is given by taking $\mathcal{H}$ to be $\mathcal{R}_1$, and can be rewritten in the familiar form found in texts on elementary catastrophe theory, such as that by Poston and Stewart [21, p. 134 and p. 159], as follows. The germ f is k-$\mathcal{R}_1$-determined if and only if

$$\mathcal{M}_n^{k+1} \subset L\mathcal{R}_1 \cdot f = \mathcal{M}_n^2\langle \partial f/\partial x_1, \dots, \partial f/\partial x_n\rangle.$$

This provides a practical criterion for $\mathcal{R}$-determinacy.

2.2. *Working with jet-groups: Mather's lemma and the detection of moduli*

The method of CTs gives a complete set of representatives for the $J^{k+1}\mathcal{G}$-orbits over a given k-jet f. This set is given as an affine space in $J^{k+1}(n, p)$ through f and we wish to reduce it further, preferably to a finite set of representatives. This can often be achieved using 'scaling' coordinate changes in the source and target, a simple problem involving linear algebra. However, in cases where moduli are present, scaling is not possible, and we need a criterion to detect such moduli. Alternatively, the family given by the affine space may be $\mathcal{G}$-trivial, collapsing to give one normal form, f. The space L used in CT calculations is generally smaller than the tangent space to the whole group $L\mathcal{G}$, so it is not surprising that a CT may contain redundant terms. (In general we cannot take L to be the whole of $L\mathcal{G}$.)

In cases where further simplification is necessary we have to work with the whole group $\mathcal{G}$, and a result specifically intended to deal with such questions is *Mather's lemma* [**25**, Lemma 1.1]. We state it in our special case of interest, where a jet-group $J^k\mathcal{G}$ acts on $J^k(n, p)$.

Lemma 2.7. *Let X be a connected submanifold of $J^k(n, p)$. Then X is contained in a single orbit of $J^k\mathcal{G}$ if and only if*

(i) *for each jet $x \in X$, the tangent space $T_x X \subset T_x(J^k\mathcal{G} \cdot x)$, and*
(ii) $\dim T_x(J^k\mathcal{G} \cdot x)$ *is constant for all $x \in X$.*

The tangent space to the orbit through x is given by the action of the Lie algebra thus, $T_x(J^k\mathcal{G} \cdot x) = L(J^k\mathcal{G}) \cdot x$. The two conditions of Mather's lemma are extremely difficult to check using hand calculations but are easily dealt with by our package, taking $L = L(J^k\mathcal{G})$. Verifying the inclusion condition (i) requires little computational overhead once a basis for the tangent space has been calculated. Note that the jet passed to the package contains arbitrary parameters and represents a whole affine space in $J^k(n, p)$. Our algorithm will provide a set of exceptional values where the dimension of the tangent space may drop or the inclusion condition (i) fails. These exceptional values are stored for examination by the user after the algorithm has terminated; see Section 3.3.

A related issue is the detection of moduli. The CT process may produce an entire family of jets which are all distinct up to $\mathcal{G}$-equivalence. To prove that moduli are indeed present we use the following straightforward criterion.

Lemma 2.8. *Let W be a smooth constructible subset of the jet-space $J^k(n, p)$ and for $w \in W$ define*

$$d(w) = \dim\Big(\big(T_w(J^k\mathcal{G} \cdot w) + T_w W\big)/T_w(J^k\mathcal{G} \cdot w)\Big).$$

Then, given an integer $r \geqslant 1$, if the set $\{\, w \in W : d(w) \leqslant r - 1 \,\}$ is a constructible subset of W of smaller dimension, then every germ f with $j^k f \in W$ is of $\mathcal{G}$-modality r or greater.

Again, this is an extremely difficult condition to check using hand calculations. It may be verified easily by our package, taking $L = L(J^k\mathcal{G})$.

2.3. *Unfolding theory*

Let $F : \mathbf{F}^n \times \mathbf{F}^s, 0 \to \mathbf{F}^p \times \mathbf{F}^s, 0$ defined by $(x, u) \mapsto (f(x, u), u)$ be an unfolding of $f_0 \in \mathcal{M}_n\mathcal{E}(n, p)$. We recall the following fundamental result from unfolding theory. (The case where $\mathcal{G}$ is one of the standard Mather groups is discussed in [**17, 25**]; for the generalisation to geometric subgroups of $\mathcal{K}$ see [**9**].)

Theorem 2.9. *F is $\mathcal{G}$-versal if and only if*

$$L\mathcal{G}_e \cdot f_0 + \mathbf{F}\{\dot{F}_1, \ldots, \dot{F}_s\} = \mathcal{E}(n, p),$$

where the initial speeds $\dot{F}_i \in \mathcal{E}(n, p)$ of F are defined by

$$\dot{F}_i(x) = \partial f/\partial u_i(x, 0), \quad \textit{for} \quad i = 1, \ldots, s.$$

Corollary 2.10. *If $g_1, \dots, g_s \in \mathcal{E}(n, p)$ form an* $\mathbf{F}$*-spanning set for the normal space to $L\mathcal{G}_e \cdot f_0$ in $\mathcal{E}(n, p)$ then $F(x, u) = (f(x) + \sum_{i=1}^{s} u_i g_i(x), u)$ is a versal unfolding of f_0, where $u = (u_1, \dots, u_s)$.*

Thus, to calculate a versal unfolding of f_0 we need to determine the g_i. As stated, this is a problem at the germ level. However, if f_0 is k-$\mathcal{G}$-determined then by the characterisation of determinacy given in [6] (see [6, Theorem 1.9] for $\mathcal{G}$ a standard Mather group, and [6, Theorems 4.5 and 4.6] for more general subgroups of $\mathcal{K}$) we have $\mathcal{M}_n^{k+1}\mathcal{E}(n, p) \subset L\mathcal{G} \cdot f_0$. But the latter is a subset of $L\mathcal{G}_e \cdot f_0$ and it therefore suffices to calculate the normal space to $L\mathcal{G}_e \cdot f_0$ in $J^k(n, p)$. This is a simple application of the package, taking $L = J^k(L\mathcal{G}_e)$. (Note that in practical situations one usually establishes k-determinacy of f_0 by applying a determinacy result such as Theorem 2.4. In this case the above inclusion $\mathcal{M}_n^{k+1}\mathcal{E}(n, p) \subset L\mathcal{G} \cdot f_0$ follows directly from the determinacy criterion anyway.)

3. *Package overview*

3.1. *Survey of existing computer packages*

We begin this section by describing several existing computer packages which are aimed at singularity theory. The packages related most closely to ours are the `CATFACT` package developed by Cowell and Wright [8]; the `OCRM` program written by Olsen, Carter and Rockwood (published in the book by Poston and Stewart [21], and corrected and enhanced by Millington [18]); and the `TGf` program written by Ratcliffe and referred to in [22]. The first two deal with the case of function-germs under $\mathcal{R}$-equivalence (an area which is often called ‘elementary catastrophe theory’). The program developed by Ratcliffe is notable in that it performs similar calculations to `Transversal` and was written, independently, at about the same time that `Transversal` was written. The original version was restricted to $\mathcal{A}$-equivalence of map-germs from surfaces to 3-space and was used successfully in [22]. Both `TGf` and `OCRM` suffered from being written in a non-symbolic language (respectively, `Pascal` and a version of `ALGOL`). The final version of `TGf` (1994) was rewritten in `Maple` and the restriction to map-germs from surfaces to 3-space was lifted. All three programs are no longer being developed. The major improvements `Transversal` makes on these packages include an extensive broadening of the types of problems considered (for example, a greater variety of equivalence relations; extensions to multigerms and lowerable fields) together with the implementation of the latest classification techniques [4, 6]. Its success in several important projects (cited in the introduction) is an indication of these claims.

We should add that the `CATFACT` package performs a lot more than determinacy and unfolding calculations. It contains a ‘recognition’ algorithm which identifies if a given function-germ belongs to one of those on Arnold’s list of low-modality singularities [1], and a ‘reduction’ algorithm which solves the ‘mapping-problem’ for unfoldings [8]. The ‘recognition’ algorithm calculates the Boardman symbol of the singularity (using Gröbner basis methods) and uses the fact that this identifies the low-modality singularities. One needs to know Arnold’s classification in advance to exploit such observations, which is why it is necessary to obtain similar classifications of map-germs under the other important equivalence relations (in particular, the $\mathcal{A}$ and $\mathcal{K}$ cases). On a similar theme, we note the ‘recognition’ program of Tari [24]. This implements a version of Arnold’s ‘determinator’ algorithm [1] using `Maple`.

Other packages aimed at singularity theory include `Singular` and `Macaulay` [11, 2], though the latter deals more with applications in algebraic geometry. Both represent extensive ongoing projects. Each has its own kernel, which is purpose-written to exploit techniques from computational commutative algebra, and its own user-interface and programming language. They have numerous applications in singularity theory, algebraic geometry and commutative algebra, but are not suited to the specialist area of classification problems discussed in this article, especially in the case of $\mathcal{A}$-classification.

3.2. *Basic functionality and underlying algorithm*

The main principle behind the algorithm is to treat the spaces $L \cdot f$ as vector subspaces of the jet-space. Once a basis has been found, we can answer all of the questions raised by the theory. For example, given some subgroup $\mathcal{G}$ of $\mathcal{K}$, whose action defines the required equivalence relation, the package can:

(i) calculate complete transversals;

(ii) check determinacy criteria;

(iii) calculate tangent spaces;

(iv) calculate codimension and versal unfoldings;

(v) check the hypotheses of Mather's lemma;

(vi) detect the presence of moduli.

Note that all of these calculations may be reduced to finite-dimensional problems within some jet-space. Cases (i), (v) and (vi) deal implicitly with calculations in a jet-space. For (ii) we appeal to results such as Nakayama's lemma or Theorem 2.5. Provided that the germ in question is finitely determined, calculations (iii) and (iv) may also be performed in an appropriate jet-space (for example, one of degree equal to the order of determinacy); see Section 2.3.

For (i) and (ii) we would typically perform the calculations using some unipotent subgroup of $\mathcal{G}$; for (v) and (vi) we work with $\mathcal{G}$ and set $L = L(J^k\mathcal{G})$, and similarly for (iii) and (iv), except that the 'extended tangent space' $L = J^k(L\mathcal{G}_e)$ may be required instead. An important consideration therefore is how the user should specify the variety of different types of spaces L which are needed in such calculations. The approach we adopt is to decompose L into the direct sum of two components: 'source' and 'target'. The source component is defined to be an $\mathcal{E}_n$-module generated by a set of user-specified vector fields. The target component is more rigid, being limited to the $\mathcal{L}$ and $\mathcal{C}$ types of equivalence at present. Several global 'setup' variables are used to specify features such as the 'type' of equivalence, the generating set of 'source' vector fields, the powers to which the maximal ideals are to be raised in the defining equation for L, and extra 'nilpotent' terms. These details are discussed further below under the section 'Initialisation Step'; see also the Transversal User Manual [**16**, Chapter 4, Section 4.2]. The scheme clearly has its limitations. However, a reasonable compromise is reached, in that virtually every case which comes up in applications is covered and, from a practical viewpoint, it is straightforward for the user to apply the setup procedure.

Assuming that the formalities of how the user actually specifies the space L have been dealt with, it is a simple matter to calculate a spanning set for L — reducing this to a basis is the major computational problem. Elements of the jet-space correspond to truncated polynomial vectors over the field of real or complex numbers. By extracting monomial coefficients we can treat jets as familiar coordinate vectors and reduce the spanning set to a basis using Gaussian elimination. A major concern with this approach is the size of the matrices involved. However, these matrices are highly sparse and, as numerous examples demonstrate, can be reduced relatively quickly. In addition, there are several features of the problem which we may exploit to reduce the computational overhead at the elimination stage. It is wasteful to extract coefficient vectors (which are generally of a high dimension) and create a matrix. Instead, we apply the elimination directly to the polynomial vectors, manipulating them as symbolic expressions. This technique will be called *indexed Gaussian elimination*. The symmetry present in the 'target' tangent spaces (for example, types $\mathcal{L}$ and $\mathcal{C}$) is exploited at the elimination stage also. We will discuss these and other technical issues in Section 3.3.

Our concerns regarding large coefficient matrices were also noted in [**18**], and Gröbner basis methods were used in the underlying algorithm in CATFACT. Although successful, this approach cannot generalise to $\mathcal{A}$-calculations because the algebraic structure of the $\mathcal{A}$-tangent space is that of a mixed module. That one has to treat the $\mathcal{A}$-tangent space as a vector space and work with the associated large matrices appears to be an unavoidable problem. The utility of our approach is ultimately measured by its success in dealing with important problems.

The stages of the algorithm are summarised below.

Initialisation step. Firstly, L is specified as one of five broad 'types', which we will denote by $\mathcal{R}$, $\mathcal{L}$, $\mathcal{C}$, $\mathcal{A}$ and $\mathcal{K}$. The required 'type' is set by a global variable, which may take the string constant values R, L, C, A and K. For 'type' $\mathcal{R}$, L is defined to act on a given jet f by

$$L \cdot f = \mathcal{M}_n^{t_1}\langle \xi_1 \cdot f, \ldots, \xi_s \cdot f\rangle,$$

where the exponent t_1 is given by a user-defined integer variable and the ξ_i are user-defined vector fields. The ξ_i are defined via a procedure which takes f as a parameter and returns the vectors $\xi_i \cdot f$; the procedure is pointed to by another global variable and is called at run-time. Several procedures are provided: the standard $\mathcal{R}$ case, where $\xi_i = \partial/\partial x_i$, is covered, as are cases where L is the space of vector fields tangent to a discriminant variety. Thus, 'type' $\mathcal{R}$, with $\xi_i = \partial/\partial x_i$ and $t_1 = 0, 1$ and 2, defines the tangent spaces $L\mathcal{R}_e$, $L\mathcal{R}$ and $L\mathcal{R}_1$, respectively. For 'type' $\mathcal{L}$, L is defined to act by

$$L \cdot f = f^*(\mathcal{M}_p^{t_2})\{e_1, \ldots, e_p\},$$

where the e_i are the canonical basis vectors in $\mathbf{F}^p$ and t_2 is a user-defined integer variable. For 'type' $\mathcal{C}$ the action is defined by

$$L \cdot f = \mathcal{M}_n^{t_2} f^*(\mathcal{M}_p)\mathcal{E}(n, p).$$

As one would expect, for 'types' $\mathcal{A}$ and $\mathcal{K}$, $L \cdot f$ is defined as the sum of the spaces defined by 'types' $\mathcal{R}$, $\mathcal{L}$ and $\mathcal{R}$, $\mathcal{C}$, respectively.

This approach allows one to define a wide range of tangent spaces L, and covers virtually everything which arises in practice. For complete transversal and determinacy techniques we often work with a unipotent subgroup of $\mathcal{K}$ and the corresponding nilpotent tangent space L is given by the sum of a 'standard' tangent space and a linear space spanned by a set of 'extra' vectors. For example, in the $\mathcal{A}$ case the space L is given by the sum of $L\mathcal{A}_1$ and a space spanned by 'extra' vectors belonging to $L\mathcal{A} \setminus L\mathcal{A}_1$; see [**4, 6**]. Further global variables specify these 'extra' vectors, and the package can be used in such situations.

Having initialised the calculation, we now call the appropriate functions in the package. The first three stages of the algorithm form the major part of the calculation, and are performed by one function which takes a jet f and jet-space degree k as parameters.

Step 1. For the given jet f, jet-space degree k and tangent space L, calculate $L \cdot f$ in $J^k(n, p)$. Specifically, calculate a spanning set for $L \cdot f$ as a vector subspace of $J^k(n, p)$. The algorithm constructs this set using the definition of $L \cdot f$ given above for each 'type', and essentially follows the same procedure as that used if one were doing the calculation by hand. For example, in a standard $\mathcal{R}$ classification, using the complete transversal method with $L = L\mathcal{R}_1$ say, we calculate $L \cdot f = \mathcal{M}_n^2 \langle \partial f/\partial x_1, \dots, \partial f/\partial x_n \rangle$ by first obtaining the vectors which generate $L \cdot f$ as an $\mathcal{E}_n$-module, $\{\partial f/\partial x_i\}$. These are multiplied by all monomials of degree 2 and higher in the source variables until we obtain jets whose initial degree is greater than the jet-space degree k. The space $J^k(n, p)$ is identified with the space of p-tuples of polynomials in n indeterminates over $\mathbf{F}$, truncated to degree k. The spanning set is therefore given as a set of such polynomial vectors.

Step 2. The spanning set calculated in Step 1 is reduced to echelon form using Gaussian elimination. By ordering the monomial vectors $x_1^{i_1} \dots x_n^{i_n} e_j \in J^k(n, p)$, each jet in $J^k(n, p)$ corresponds to a coordinate vector over $\mathbf{F}$ via extraction of coefficients. The spanning set obtained in Step 1 then corresponds to the matrix whose rows consists of these coefficient vectors. Reducing this matrix using Gaussian elimination gives a canonical basis for the tangent space. We actually use the technique of indexed Gaussian elimination, mentioned above and discussed in Section 3.3.

Step 3. A basis C for the complementary (normal) space to the tangent space is calculated. That is, the independent set obtained in Step 2 is extended to one of full rank in $J^k(n, p)$ by the addition of monomial vectors. In the $\mathcal{A}_e$ and $\mathcal{A}$ cases (for example) this gives the terms required in a versal unfolding and the corresponding codimension (for determined jets). In the $\mathcal{A}_1$ complete transversal case (for example) the monomial jets in $J^k(n, p)$ are ordered so that those of degree k correspond to the latter columns of the matrix. The monomial vectors in C of degree k will then form a basis for a complete transversal. This process can be generalised to deal with complete transversal calculations using a unipotent subgroup $\mathcal{G}$ and corresponding nilpotent filtration; see Section 3.3.

Step 4. Calculating a basis for the tangent space is the main computational overhead in the algorithm. During this procedure all by-products of the reduction process which may be of further use (such as the bases for the tangent and normal spaces, invariants such as the dimension and codimension of these spaces) are stored as global variables for access by other routines. Step 4 deals with output and manipulation of these results. A number of procedures are associated with Step 4 and perform functions such as displaying the bases, displaying a basis for a complete transversal, and testing whether a given set of vectors is independent to the tangent space (such questions arise in checking the hypotheses of Mather's Lemma and in moduli detection). The computational overhead of such procedures is negligible compared to that involved in Steps 1 – 3.

3.3. *Technical and computational considerations*

We will now describe some of the more important computational issues behind the algorithm. Further details on the actual program code and a presentation of parts of the algorithm in the form of pseudo code were given in [**15**]. In addition, we remark that the `Maple` source code is fully documented.

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

3.3.1. *Symbolic pivots: fraction-free Gaussian elimination*

Writing the package in a symbolic language such as `Maple` allows great flexibility. One notable advantage is that parameters (such as moduli) may be present in the jets we work with, thus allowing us to perform calculations for whole families. The matrix created in Step 2 will contain polynomial entries, and we must take this into account during the Gaussian elimination routine. We choose numeric pivotal elements (in this context meaning 'constant polynomials') where possible, but when we are forced to choose a non-constant polynomial pivotal element *no* division is performed on the chosen row to reduce the pivot to unity. Division is still performed (working in the field of rational functions) when using the pivot to reduce the rest of the column to zero. This is in contrast to standard 'fraction-free' Gaussian elimination [**10**, p. 82] where the pivot and the term it is to eliminate are multiplied up, and no division occurs at all. Our method provides a valid elimination algorithm for jets involving parameters without the inconvenience of standard fraction-free elimination where the matrix entries rapidly blow-up into large expressions. The elimination only breaks down for certain values of the parameters for which a pivot vanishes, but the conditions determining this are retained. The list of all non-numeric pivots is stored for global access after the algorithm terminates, and may be examined by a procedure associated with Step 4.

The non-numeric pivots will, in general, be rational functions in the parameters, the vanishing of their numerators defining a finite set of proper algebraic varieties within the parameter space. The elimination applies to members of the family corresponding to values of the parameters not lying on these varieties, and the algorithm therefore determines the generic behaviour by default. To investigate the exceptional behaviour we must inspect each of the non-numeric pivots in turn, obtaining conditions on the parameters for which the elimination breaks down. In many cases (at least those with one or two parameters) the solutions can be determined explicitly using one of the `Maple` factor or solver procedures, the solutions being substituted back into the family and the calculation repeated. This process detects phenomena such as exceptional values in modular families, or cases where applications of Mather's lemma break down thus obstructing triviality within the family but providing a finite list of normal forms.

3.3.2. *Exploiting sparsity: indexed Gaussian elimination*

Working with a matrix of coefficient vectors in Step 2 is wasteful on memory and CPU time. By the very nature of the algorithm, the data is created as a set of polynomial vectors (truncated to the prescribed degree k). This is a very efficient data structure to work with. Storage of the sparse data (the non-zero coefficients) is optimised, as is its manipulation. The idea is to work with the set of polynomial vectors and manipulate these directly using symbolic techniques; thus a coefficient matrix is never created. We use a set of indexing tables which, for a given row and column (i, j) of the would-be coefficient matrix, index the appropriate coefficient of the ith polynomial vector in our spanning set. The column j therefore indexes two pieces of information: the component of the vector and a monomial term in the resulting polynomial. During elimination, coefficients are looked-up from this set of polynomial vectors using the indexing tables and, for all intents and purposes, could be thought of as matrix entries. However, the row-reduction operations performed in Gaussian elimination are now achieved by direct polynomial addition — a very efficient process in `Maple`, which uses the internal kernel functions.

We had to completely rewrite the Gaussian elimination routine found in `Maple` in order to incorporate both the above method and the type of fraction-free elimination described in the previous section. The resulting elimination algorithm proved, on average, to be two

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

to three times faster, using three to four times less memory than methods that extract an explicit matrix of coefficients and apply the standard `Maple` library routines.

3.3.3. *Exploiting symmetry in the target*

The $\mathcal{L}$ and $\mathcal{C}$ 'type' tangent spaces exhibit a large degree of symmetry. Their respective action on a given germ f is given by,

$$f^*(\mathcal{M}_p^{t_2})\{e_1, \ldots, e_p\} \qquad \text{and} \qquad \mathcal{M}_n^{t_2} f^*(\mathcal{M}_p)\mathcal{E}(n, p).$$

In the $\mathcal{L}$ case we create a spanning set for the ideal $f^*(\mathcal{M}_p^{t_2})$ as a vector subspace of $J^k(n, 1)$ and reduce this to echelon form using (indexed) Gaussian elimination. In the $\mathcal{C}$ case we do the same for the ideal $\mathcal{M}_n^{t_2} f^*(\mathcal{M}_p)$. We then produce a spanning set for the full $\mathcal{L}$ or $\mathcal{C}$ tangent space by stacking together p copies of the resulting 'matrix'. The important point is that we can do this in such a manner as to create a spanning set for the full tangent space which is *already* in echelon form and therefore requires no further elimination. This is clear if the matrices were stacked together to form a diagonal block matrix, but this corresponds to a specific ordering of the monomial vectors in $J^k(n, p)$. The monomial orderings required in certain problems, such as complete transversal calculations, do not give rise to such a simple diagonal block matrix, but the principle still applies and we indeed find that the matrix formed by stacking is automatically in echelon form. The reduction in computational overhead is clear.

Finally, in a problem dealing with $\mathcal{A}$-equivalence or $\mathcal{K}$-equivalence, this basis for the target tangent space is adjoined with a spanning set for the source tangent space and the resulting 'matrix' reduced to echelon form. Represent these as matrices of coefficient vectors, M_1 and M_2, respectively. The full tangent space matrix, formed by adjoining these,

$$\begin{pmatrix} M_1 \\ M_2 \end{pmatrix}$$

is reduced to echelon form. However, M_1 is already in echelon form and a full-blown Gaussian elimination is replaced by the following algorithm. Keep the current row and column pointer in the matrix M_1. If the corresponding entry is a pivot then reduce as usual; only the column in M_2 needs to be reduced to zero, as the column in M_1 will already be zero. Otherwise, (if the entry in M_1 is zero) try and find a pivot in M_2. If this is possible, again only the column in M_2 needs to be reduced. However, if we need to use M_2 to obtain a pivot then we do not *swap* the rows of M_1 and M_2 as in standard Gaussian elimination, but rather *insert* the row of M_2 into M_1, thus preserving the fact that M_1 is echelon. This is the basic idea at least. In the code it is more efficient to create a separate matrix that stores the final result: when a pivot is found the corresponding row is added to this 'result matrix', thus eliminating the need to insert a row of M_2 into M_1 (moving all the remaining rows of M_1 down). In addition, the process is carried out using the indexing tables referred to above, not coefficient matrices.

We remark that the presence of target tangent spaces and a target dimension greater than 1 make the computational overhead at the elimination stage *considerably* greater in $\mathcal{A}$- and $\mathcal{K}$-'type' calculations, compared to $\mathcal{R}$-'type' calculations. This exploitation of symmetry means that many significant calculations remain feasible.

3.3.4. *Normal spaces, complete transversals and nilpotent filtrations*

It is an easy matter to extend the basis for $L \cdot f$ to one of full rank in $J^k(n, p)$, thus providing a basis C for the normal space. If the basis for $L \cdot f$ is given in coordinate form by the rows of the echelon matrix (a_{ij}) with pivotal elements $a_{1j_1}, a_{2j_2}, \dots, a_{rj_r}$ (so these are non-zero elements and $1 \leqslant j_1 < j_2 < \cdots < j_r \leqslant q = \dim J^k(n, p)$, where $r = \dim L \cdot f$), then the canonical vectors

$$\{e_1, \dots, \hat{e_{j_1}}, \dots, \hat{e_{j_2}}, \dots, \hat{e_{j_r}}, \dots, e_q\}$$

(where $\hat{e_j}$ denotes the exclusion of e_j from this set of vectors) form the basis C. Of course, we calculate C as the set of monomial vectors which correspond to these coordinate vectors e_i. The algorithm to derive C from (a_{ij}) also takes the opportunity to pick off all of the non-numeric pivots (discussed above) and store them for global access.

Calculating complete transversals requires a little more subtlety. Provided that the columns corresponding to the monomial jets of degree k appear as a block at the end of the column, the above procedure will provide a basis for a degree k complete transversal associated with the standard filtration by degree (see Example 2.3). This basis simply consists of those elements in C of degree k. For this to work for the general complete transversal Theorem 2.1 we must order the degree k monomial jets according to the nilpotent filtration, starting with those of degree $(k, 1)$, then those of degree $(k, 2)$, and so on. In most situations that arise in practice, this can be achieved via a system of weights. In what follows $\alpha = (\alpha_1, \dots, \alpha_n)$ and $\beta = (\beta_1, \dots, \beta_p)$ will denote the source and target weights respectively. We recall the following; see [4, Section 2.3] for a full discussion on weighted filtrations. The monomial vector $x_1^{k_1} \dots x_n^{k_n} e_i$ is assigned a weight $k_1\alpha_1 + \cdots + k_n\alpha_n - \beta_i$. The $\mathcal{E}_n$-submodule of $\mathcal{M}_n\mathcal{E}(n, p)$ generated by such monomial vectors of weight $\geqslant k$ is denoted $F^k_{\alpha,\beta}\mathcal{E}(n, p)$.

We consider the case of $\mathcal{A}$-classification, though the method extends to other subgroups of $\mathcal{K}$. Let $(x_1, \dots, x_n)$ denote coordinates on $(\mathbf{F}^n, 0)$ and $(y_1, \dots, y_p)$ those on $(\mathbf{F}^p, 0)$. Let $\mathcal{G}$ be a subgroup of $\mathcal{A}$ such that $L(J^1\mathcal{G})$ acts nilpotently on $\mathbf{F}^{n+p}$. For 'large enough $\mathcal{G}$' (we make this precise below) we can assign source and target weights such that the partition of the monomial vectors of (standard) degree k via their weight corresponds to their partition into the (k, s)-jet-levels using the nilpotent filtration. The following proposition was proved in [4].

Proposition 3.1. *Suppose that $L\mathcal{G}$ contains the following vectors and assign source and target weights according to the case in question.*

Vectors		*Weight*
$x_i\partial/\partial x_{i+1} \in L\mathcal{R}$	*or*	$\alpha = (n, \dots, 2, 1)$
$x_{i+1}\partial/\partial x_i \in L\mathcal{R}$		$\alpha = (1, 2, \dots, n)$
$y_j\partial/\partial y_{j+1} \in L\mathcal{L}$	*or*	$\beta = (0, -1, \dots, -p+1)$
$y_{j+1}\partial/\partial y_j \in L\mathcal{L}$		$\beta = (-p+1, \dots, -1, 0)$

for $i = 1, \dots, n-1$ and $j = 1, \dots, p-1$. Then

$$\sum_{i \geqslant s} (L\mathcal{G})^i \cdot (\mathcal{M}_n^k \mathcal{E}(n, p)) + \mathcal{M}_n^{k+1}\mathcal{E}(n, p) =$$

$$\left(F^{k+s}_{\alpha,\beta}\mathcal{E}(n, p) \cap \mathcal{M}_n^k\mathcal{E}(n, p)\right) + \mathcal{M}_n^{k+1}\mathcal{E}(n, p).$$

So for fixed k, the $M_{k,s}(\mathcal{G})$ filtration can be replaced by the weighted filtration modulo $\mathcal{M}_n^{k+1}\mathcal{E}(n, p)$; that is, the filtration on the right-hand side of the above expression. In particular, the homogeneous monomial vectors of degree (k, s) (to be precise, those that span the space given by the image of $M_{k,s-1}(\mathcal{G})$ in the jet-space $J^{k,s}(n, p)$) are just the homogeneous monomial vectors of (standard) degree k with weight $k + s - 1$.

The vectors referred to in Proposition 3.1 are the 'extra' vectors present in $L\mathcal{A} \setminus L\mathcal{A}_1$. For classification purposes one would prefer to use some unipotent group $\mathcal{G}$ such that the nilpotent Lie algebra $L\mathcal{G} \subset L\mathcal{A}$ contains as many of these 'extra' vectors as possible. There are four natural cases to consider:

$$L\mathcal{G} = L\mathcal{A}_1 \oplus \mathbf{F}\{x_i\partial/\partial x_j\} \oplus \mathbf{F}\{y_k\partial/\partial y_l\}$$

for all $i < j$ (or alternatively all $i > j$) and similarly for k and l. Such cases are used in practical applications (such as the examples in Section 4), and Proposition 3.1 applies.

4. *Examples and comments*

We will demonstrate the utility of our package via several examples. Only the results are described; a detailed tutorial on how to use the package to perform such calculations, giving a summary of user input and corresponding computer output, is described in the Transversal User Manual [**16**].

Firstly, we concentrate on the $\mathcal{A}$-classification of corank-1 map-germs $\mathbf{F}^2, 0 \to \mathbf{F}^2, 0$ having $\mathcal{A}$-codimension less than or equal to 6 ('codimension' refers to that of a stratum when moduli feature). Such germs represent the (corank 1) singularities which occur in versal 4-parameter families and have geometrical applications in areas such as computer vision, representing the profiles (apparent contours) of smooth surfaces. The classification is due to Rieger, though the cases of lower codimension should rightly be accredited to several people, beginning with Whitney; see [**23**].

The $\mathcal{A}$-classification of map-germs $\mathbf{F}^3, 0 \to \mathbf{F}^4, 0$ found in [**14**] represents the most extensive application of our package to date. A complete classification of the corank-1 simples, and those of $\mathcal{A}$-codimension less than or equal to 7 (equivalently, for the non-stables, $\mathcal{A}_e$-codimension $\leqslant 4$), is given, and we describe some of these results below. The motivation for this work was to provide more examples for the topological study of $\mathcal{A}$-finite map-germs. Mond's $\mathcal{A}$-classification of map-germs from surfaces to 3-space proved to be valuable in developing much of this theory. However, more examples were needed to test generalisations and conjectures of the theory, and map-germs $\mathbf{F}^3, 0 \to \mathbf{F}^4, 0$ were the clear candidates.

Finally, we discuss the performance of our package. We will concentrate on some of the more computationally demanding calculations, labelling these C1, C2, ..., C5 as they are encountered in the examples.

4.1. $\mathcal{A}$-*classification of map-germs* $\mathbf{F}^2, 0 \to \mathbf{F}^2, 0$

We cannot describe all of the classification in this small section; rather, we will concentrate on the more difficult branches, beginning with the 2-jet $(x, 0)$. The results apply to both the real and complex cases, apart from a few minor differences which are described as necessary. (From now on a and b will denote real or complex numbers, and be used as parameters for the affine spaces given by the complete transversal theorem. We will recycle this notation from case to case to save space.) All determinacy and CT calculations will use the unipotent subgroup $\mathcal{G}$ of $\mathcal{A}$ defined in Example 2.2.

A (3, 1)-CT for $(x, 0)$ is (x, ay^3). Apply scaling coordinate changes to reduce this family to the two cases (x, y^3) and $(x, 0)$, depending on whether $a \neq 0$ or $a = 0$. In the first case the only non-empty higher $(3, s)$-CT is at the (3, 3) level; this gives the family $(x, y^3 + ax^2y)$ which again reduces to two cases via scaling. Returning to the (3, 1)-jet $(x, 0)$, a (3, 2)-CT is (x, axy^2) and scaling reduces this to the two cases (x, xy^2) and $(x, 0)$. For the first

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

of these all of the higher $(3, s)$-CTs are found to be empty, while in the second case the only non-empty CT is the $(3, 3)$-CT giving (x, ax^2y). After further scaling we arrive at the complete list of 3-jets over $(x, 0)$, namely

$$(x, y^3 + x^2y), \quad (x, y^3), \quad (x, xy^2), \quad (x, x^2y), \quad (x, 0).$$

A simple computer calculation shows that these have $J^3\mathcal{A}$-codimension 3, 4, 4, 5 and 6, respectively. (These codimension calculations are very quick and easy using our computer package. They provide useful invariants which automatically distinguish many $J^k\mathcal{A}$-types and help one to recognise the $\mathcal{A}$-type of a given map-germ by following the appropriate branches up the classification tree.) We remark that the above classification of 3-jets over $(x, 0)$ may be obtained via other methods, but this generally requires many ad-hoc techniques. The above example demonstrates the practicality of our classification techniques, which apply in the same straightforward manner in different situations.

We continue the classification, taking the 3-jet (x, x^2y) as our example. Using similar arguments to those given, we find that a $(4, 1)$-CT reduces to give the two cases (x, x^2y+y^4) and (x, x^2y). For the first, the higher $(4, s)$-CTs are empty, while for the second the $(4, 2)$-CT reduces to give the two cases $(x, x^2y + xy^3)$ and (x, x^2y). The complete list of 4-jets over (x, x^2y) is therefore

$$(x, x^2y + y^4), \quad (x, x^2y + xy^3), \quad (x, x^2y),$$

having $J^4\mathcal{A}$-codimension 5, 6 and 7, respectively.

The only non-empty 5-CT for $(x, x^2y + y^4)$ produces the family $(x, x^2y + y^4 + ay^5)$. With a little linear algebra one can show that scaling coordinate changes may reduce the above family to the three orbits $(x, x^2y + y^4 \pm y^5)$ and $(x, x^2y + y^4)$. However, it is far more straightforward to apply Mather's lemma using the computer. A very quick calculation with $f_a = (x, x^2y + y^4 + ay^5)$ verifies that $(0, y^5) \in L(J^5\mathcal{A}) \cdot f_a$ and $L(J^5\mathcal{A}) \cdot f_a$ has codimension 5 in $J^5(2, 2)$ provided that $a \neq 0$. (That is, the matrix reduction required the use of a symbolic pivotal element having a as a factor and this was noted by the package; see Section 3.3.) Repeating the calculation for $a = 0$ we find that the codimension increases to 6 (and the inclusion condition fails). Thus, by Mather's lemma, we obtain the three orbits listed above and note that the first two have $J^5\mathcal{A}$-codimension 5, the latter 6. The $\pm$ sign appears due to the 'connectedness' condition in Mather's lemma. Of course, over **C** this does not feature, and the first two orbits are $\mathcal{A}$-equivalent. Distinguishing the $\pm$ orbits over **R** is a problem. One usually hopes to find invariants that will do this, but generally such questions cannot be answered using the techniques discussed here. Continuing the classification, we find that in all cases the CTs of degree 6 to degree 9 are empty. This implies that $\mathcal{M}_2^6\mathcal{E}(2, 2) \subset L\mathcal{G} \cdot f + \mathcal{M}_2^{10}\mathcal{E}(2, 2)$ where f denotes any of the above jets. Since $f^*(\mathcal{M}_2)\mathcal{E}(2, 2) \supset \mathcal{M}_2^4\mathcal{E}(2, 2)$ Theorem 2.5 shows that these jets are all 5-determined. We will discuss the computational aspects of the $J^9(2, 2)$ calculation for $(x, x^2y + y^4 + y^5)$ (that is, the verification that the largest degree CT is empty), and denote this calculation by C1 for future reference.

We return to the 4-jet $(x, x^2y + xy^3)$. A 5-CT is $f_a = (x, x^2y + xy^3 + ay^5)$. Scaling coordinate changes cannot simplify this family but this does not *prove* that a is a modulus. For this we apply Lemma 2.8, and computer calculation verifies that $(0, y^5) \notin L(J^5\mathcal{A}) \cdot f_a$ for all a. In addition, the computer shows that any representative of the stratum formed by this unimodular family has $J^5\mathcal{A}$-codimension 7. (Although we are working up to $\mathcal{A}$-codimension 6, the stratum as a whole has codimension 6.) Continuing, a 6-CT is $f_{a,b} = (x, x^2y + xy^3 + ay^5 + by^6)$. We apply Mather's lemma to try and simplify this family.

Computer calculation shows that $(0, y^6) \in L(J^6\mathscr{A}) \cdot f_{a,b}$, the orbit having codimension 7, for all a and for $b \neq 0$. For $b = 0$ (and all a) the codimension jumps to 8. We therefore obtain the two orbits

$$(x, x^2y + xy^3 + ay^5 + y^6), \quad (x, x^2y + xy^3 + ay^5),$$

a representative of these unimodular families having $J^6\mathscr{A}$-codimension 7 and 8, respectively. The latter exceeds the codimension bounds, so we just consider the former. (Note that in the real case, the former contains a $\pm y^6$ term, but this can be reduced to $+y^6$ using a simple coordinate change.)

A 7-CT is $f_{a,b} = (x, x^2y + xy^3 + ay^5 + y^6 + by^7)$ (for all a except $a = 3/2$ where the CT is empty). Computer calculation shows that the vectors $(0, y^5)$ and $(0, y^7)$ span an independent set to $L(J^7\mathscr{A}) \cdot f_{a,b}$ for generic (a, b), so by Lemma 2.8 $f_{a,b}$ is a *bimodular* family. A generic representative of this family has $J^7\mathscr{A}$-codimension 8. By 'generic' we mean that (a, b) does not lie on a finite set of proper algebraic varieties in $\mathbf{F}^2$. These varieties can be determined by computer; in this case they are given by simple conditions such as $a = 3/2$ (where the family simplifies as shown already by the CT calculation) or $(a, b) = (9/5, -4/3)$ where the $J^7\mathscr{A}$-codimension of the jet jumps to 9. A full analysis of the exceptional values of the moduli requires extra investigation but is fairly straightforward using the computer. Note that any non-generic strata will be of too high a codimension.

Continuing, we find that the 8-CT is empty for generic (a, b); the exceptional conditions include $a = 0$ (where another modulus appears) and we assume that $a \neq 0$ from now on. Further calculations show that all CTs from degree 8 to degree 12 are empty for generic (a, b). Thus, $\mathscr{M}_2^8\mathscr{E}(2, 2) \subset L\mathscr{G} \cdot f_{a,b} + \mathscr{M}_2^{13}\mathscr{E}(2, 2)$, and since $a \neq 0$ we have $f_{a,b}^*(\mathscr{M}_2)\mathscr{E}(2, 2) \supset \mathscr{M}_2^5\mathscr{E}(2, 2)$ so by Theorem 2.5 $f_{a,b}$ is 7-determined for generic (a, b). We will discuss the computational aspects of the $J^{12}(2, 2)$ calculation for $f_{a,b}$ later, and denote this calculation by C2.

Returning to the final 4-jet (x, x^2y), we note that its codimension exceeds the bounds, thus completing this branch of the classification. Although a simple example, our main aim in discussing this classification is to demonstrate how a tedious calculation may be carried out very quickly using `Transversal`. The above results can be achieved within a matter of minutes, and provide independent verification of Rieger's results. Far more complicated classifications can be achieved using `Transversal`, but the style of the approach is identical to that just described, as we will now demonstrate.

4.2. *$\mathscr{A}$-classification of map-germs* $\mathbf{F}^3, 0 \to \mathbf{F}^4, 0$

We will only discuss a few specific calculations; for a concise summary of the classification see [**14**].

Let (x, y, z) denote coordinates in the source, and (u_1, u_2, u_3, u_4) those in the target. Let $\mathscr{G}$ be the unipotent subgroup of $\mathscr{A}$ having nilpotent Lie algebra

$$L\mathscr{A}_1 \oplus \mathbf{F}\{x\partial/\partial y, x\partial/\partial z, y\partial/\partial z\} \oplus \mathbf{F}\{u_i\partial/\partial u_j \quad \text{for} \quad i > j\}.$$

This group will be used in all of the CT and determinacy calculations. Consider the jet-spaces $J^{r,s}(3, 4)$ induced by the nilpotent filtration. The monomial vectors of (standard) degree r are partitioned into their (r, s)-levels as described in Section 3.3 using the weights $\alpha = (3, 2, 1)$ and $\beta = (-3, -2, -1, 0)$.

As our first example we consider (x, y, yz, xz), which occurs as one of the five corank-1 2-jets. A $(3, 1)$-CT is $(x, y, yz, xz + az^3)$ and applying scaling coordinate changes reduces

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

this family to the two cases $(x, y, yz, xz + z^3)$ and (x, y, yz, xz). In both cases the only higher non-empty $(3, s)$-CT is the $(3, 2)$-CT, and we obtain the 3-jets $(x, y, yz+az^3, xz+z^3)$ and $(x, y, yz + az^3, xz)$. In the first case it is tempting to scale a to 0 or 1. However, the resulting 3-jets have the same $J^3\mathcal{A}$-codimension; indeed, the codimension is found to be constant for all a, suggesting that the family is $\mathcal{A}$-trivial. A simple application of Mather's lemma shows that this is so. We therefore have the one normal form: $(x, y, yz, xz + z^3)$. Returning to the second jet $(x, y, yz + az^3, xz)$, here we apply scaling coordinate changes and reduce this to the 3-jets $(x, y, yz + z^3, xz)$ (equivalent to $(x, y, yz, xz + z^3)$ obtained earlier) and (x, y, yz, xz). The complete list of 3-jets over (x, y, yz, xz) is therefore

$$(x, y, yz, xz + z^3), \quad (x, y, yz, xz),$$

having $J^3\mathcal{A}$-codimension 4 and 6, respectively. As an instructive example we consider the same calculation using, instead, the group $\mathcal{A}_1$. In this case a 3-CT over (x, y, yz, xz) is

$$(x, y, yz + a_1z^3 + a_2xz^2, xz + a_3z^3 + a_4xz^2 + a_5yz^2) \quad \text{for} \quad a_i \in \mathbf{F}.$$

Of course, one may reduce these to the two cases above, but this would involve a lot of (ad-hoc) work. To classify the 3-jets over (x, y, yz, xz) without the use of techniques such as CTs would be a very unenviable task!

Continuing the classification over the first of the above 3-jets gives the series $(x, y, yz + z^k, xz+z^3)$, k-determined for $k \geqslant 4$, k not a multiple of 3. (Note that this is not the complete classification of all jets over this 3-jet. Further branching occurs at the 6-level and 7-level. This is an important example in theoretical singularity theory, in that the 3-jet gives rise to a series but is not a stem.) We take the determinacy calculations of the first two members of this series as our example. Using Theorem 2.5 and noting that $f^*(\mathcal{M}_3)\mathcal{E}(3, 4) \supset \mathcal{M}_3^3\mathcal{E}(3, 4)$ in both cases, we establish 4-determinacy for $k = 4$ by showing that the CTs from degree 5 to degree 7 are empty. Similarly, 5-determinacy is established for $k = 5$ by showing that the CTs from degree 6 to degree 8 are empty. We will denote the $J^7(3, 4)$ calculation for $k = 4$ by C3, and the $J^8(3, 4)$ calculation for $k = 5$ by C4.

As our final example we consider the second of the above 3-jets. Classification over this jet becomes complicated; a lot of branching occurs, with the highest branch that we must consider arising at the 9-jet-level in the form of the trimodular family $(x, y, yz + xz^3 + az^6 + z^7 + bz^8 + cz^9, xz + z^4)$. (A generic representative of this family has $\mathcal{A}$-codimension 10, the whole stratum having codimension 7.) We can establish 9-determinacy of this family using Theorem 2.5 if we can show that the CTs of degree 10 to degree 13 are all empty. This calculation represents one of the most intensive determinacy calculations carried out, and will be discussed next, denoted by C5.

4.3. *Comments*

The ultimate test for our program is its ability to solve classification problems such as those described above, whether these involve simple calculations which may be done by hand, or intensive calculations which eventually require the use of a computer in some capacity anyway. The majority of the calculations described above were dealt with in seconds using our package. Even though many calculations may be performed by hand, our computer package still acts as a valuable tool, giving quick answers to the repetitive and tedious calculations one faces in this area of singularity theory. We will now discuss some of the more computationally demanding problems. We state these results only as an indication of how the package performs on the sort of machines commonly available at the present time; they are not intended as benchmarks for such calculations.

Table 2: CPU time (hours/mins/secs) and matrix dimensions for calculations.

Calc	486	PEN	IPX	SPC	ULT	Matrix Dim
C1	31s	12s	20s	09s	03s	87,110
C2	02m06s	52s	01m32s	41s	15s	153,182
C3	08m59s	03m32s	06m02s	02m55s	59s	321,480
C4	17m06s	06m43s	11m35s	05m40s	01m56s	447,660
C5	—	—	—	06h34m01s	02h29m39s	1572,2240

Calculations C1 and C2 were described in Section 4.1; calculations C3, C4 and C5 in Section 4.2. The calculations were carried out on the following machines. We give the symbol used to identify the machine, followed by its specification (machine name, processor type, processor speed, total RAM):

486 (PC, 486DX, 50 MHz, 8 MB);
PEN (PC, Pentium, 75MHz, 16 MB);
IPX (Sun IPX workstation, Sparc, 40 MHz, 32 MB);
SPC (Sun SPARCcenter 2000, Sparc ($\times$18), 50 MHz, 276 MB);
ULT (Sun Ultra 2, Sparc ($\times$2), 168 MHz, 256 MB).

The calculations were done using a standard `Maple V` Release 3 'terminal session' (as opposed to an X11 or Windows interface) running under DOS 6.2 (PCs) and SunOS 5.5 (Sun workstations).

Table 2 shows the CPU time and coefficient matrix dimensions (for the full $\mathscr{A}$-tangent space) for each of the calculations. The matrix dimensions give an indication of the complexity of the problem, and how this increases with n, p and the jet-space degree k. A natural theoretical measure of the complexity is given by the dimension of the jet-space $J^k(n, p)$. This is the column dimension of the coefficient matrix, and can be shown to equal $p\binom{n+k}{k}$. Thus, although linear in p, this grows rapidly with n and k. The need for techniques to reduce the computational overhead, as discussed in Section 3.3, is clear.

As calculations become more intensive they are best carried out on larger Unix machines (such as 'SPC') where more resources are available. However, most of the calculations performed to date did not require such hardware. All of the above machines handled the calculations C1 – C4 within an acceptable time (the amount of real time required being little more than the stated CPU time, though this may not be the case on heavily loaded multiuser systems). The calculation C5 was attempted on a powerful Silicon Graphics Challenge machine but failed, and represents a limit to our package. The problem is due to the high modality of the family and the complexity of the equations, which determine the exceptional values of the moduli. These equations are unlikely to be of use even if the calculation is completed. We therefore attempt to show that the family is finitely determined for generic (a, b, c). If we find that for a fixed value of the moduli the CTs of degree 10 to 13 are all empty then the corresponding germ is 9-determined. Since finite-determinacy is an open condition, we can conclude that the family is finitely determined for generic values of (a, b, c). In addition, should we need to consider a specific example in applications, then we have a member of the family whose exact determinacy degree (9) is known. The calculation using fixed moduli is computationally *far* less intensive. Based on previous calculations,

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press

we avoid values of the moduli which may be exceptional and try a suitable choice. (After all, if the family is generically determined then we are quite likely to choose correctly!) The case $(a, b, c) = (5, 3, 4)$ is found to be 9-determined. Calculation C5 in Table 2 represents the $J^{13}(3, 4)$ calculation for these values of the moduli; it was carried out only on the two large Unix machines.

In summary, the package performs well in most situations. The computational complexity of the problems increases significantly with the dimensions n and p and the jet-space degree k, but remains within practical limits for many problems. The largest obstruction to calculations appears to come from the presence of moduli. Examples suggest that calculations for families with 3 or more moduli become infeasible in jet-spaces of degree in the region of 10 to 20 (depending on n and p). This is an inherent problem caused by the creation of symbolic expressions during the elimination process which rapidly become large, often too large for `Maple` to handle. When this happens `Maple` will terminate the process with an 'object too large' error, and throwing more CPU time or memory at such problems is unlikely to solve them.

Acknowledgements. We would like to thank J. W. Bruce, J. N. L. Connor, C. A. Hobbs and the two referees for making helpful comments and suggestions. The author was supported by the EPSRC grants GR/J28162, GR/L33948 and GR/L17245; computational work was supported by grant GR/H59855.

Appendix A. *Additional documentation*

This appendix contains a README file (text), which gives detailed installation instructions for the package, and a comprehensive user manual (PDF). Note that these files are also available within the tar archives that contain the complete package. The material is is to be found at

http://www.lms.ac.uk/jcm/3/lms1999-002/appendix-a/.

Appendix B. *Transversal 3.1*

This appendix contains version 3.1 of the package, provided as a tar archive. It runs under `Maple` V Releases 1–4, and is to be found at

http://www.lms.ac.uk/jcm/3/lms1999-002/appendix-b/.

Appendix C. *Transversal 3.2*

This appendix contains version 3.2 of the package, provided as a tar archive. It runs under `Maple` V Release 5. Other than the changes required for compatibility with Release 5, this version provides the same functionality as version 3.1. The package is to be found at

http://www.lms.ac.uk/jcm/3/lms1999-002/appendix-c/.

References

1. V. I. Arnold, 'Critical points of smooth functions and their normal forms', *Russian Math. Surveys* 30 (1975) 1–75. 214, 214

2. D. Bayer and M. Stillman, 'Macaulay: a system for computation in algebraic geometry and commutative algebra' (1982–1994). Source and object code available from the authors for Unix and Macintosh computers: http://www.math.uiuc.edu/Macaulay2. 214

3. J. W. Bruce, 'Generic geometry and duality', *Singularities*, Lille 1991, London Math. Soc. Lecture Note Ser. 201 (ed. J.-P. Brasselet, Cambridge University Press, 1994). 208

4. J. W. Bruce, N. P. Kirk and A. A. du Plessis, 'Complete transversals and the classification of singularities', *Nonlinearity* 10 (1997) 253–275. 207, 207, 208, 209, 209, 210, 214, 216, 220, 220

5. J. W. Bruce, N. P. Kirk and J. M. West, 'Classification of map-germs from surfaces to four-space', Preprint, University of Liverpool, 1995. 208

6. J. W. Bruce, A. A. du Plessis and C. T. C. Wall, 'Determinacy and unipotency', *Invent. Math.* 88 (1987) 521–554. 207, 208, 209, 209, 210, 211, 212, 212, 214, 214, 214, 214, 216

7. B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan and S. M. Watt, *Maple V language reference manual* (Springer-Verlag and Waterloo Maple Publishing, 1991). 207

8. R. G. Cowell and F. J. Wright, 'CATFACT: computer algebraic tools for applications of catastrophe theory', Proc. EUROCAL '87, European Conference on Computer Algebra, Leipzig, GDR, 1987, Lecture Notes in Comput. Sci. 378 (ed. J. H. Davenport, Springer, 1989) 71–80. 214, 214

9. J. N. Damon, 'The unfolding and determinacy theorems for subgroups of $\mathcal{A}$ and $\mathcal{K}$', *Mem. Amer. Math. Soc.* 306 (1984). Also in *Singularities*, Proc. Symp. Pure Math. 40 (part 1) (American Mathematical Society, Providence, RI, 1983) 233–254. 208, 213

10. L. Fox, *An introduction to numerical linear algebra* (Clarendon Press, Oxford, 1964). 218

11. G.-M. Greuel, G. Pfister and H. Schoenemann, 'The SINGULAR project', Department of Mathematics, University of Kaiserslautern. Latest information available from http://www.singular.uni-kl.de. 214

12. W. Hawes, 'Multi-dimensional motions of the plane and space', Ph.D. thesis, University of Liverpool, 1994. 208

13. C. A. Hobbs and N. P. Kirk, 'On the classification and bifurcation of multigerms of maps from surfaces to 3-space', *Math. Scand.* To appear. 208

14. K. Houston and N. P. Kirk, 'On the classification and geometry of corank-1 map-germs from 3-space to 4-space', *Singularity theory*, Proceedings of the European Singularities Conference in honour of C. T. C. Wall on the occasion of his 60th birthday (ed. J. W. Bruce and D. M. Q. Mond, Cambridge University Press, 1999) 325–351. 208, 221, 223

15. N. P. Kirk, 'Computational aspects of singularity theory', Ph.D. thesis, University of Liverpool, 1993. 208, 217

16. N. P. Kirk, 'Transversal: a Maple package for singularity theory; user manual, Version 3.1', Preprint, University of Liverpool, 1998. Available with the `Transversal` package via ftp. (See also Appendix A, Appendix B and Appendix C above.) 208, 209, 215, 221

17. J. Martinet, *Singularities of smooth functions and maps*, London Math. Soc. Lecture Note Ser. 58 (Cambridge University Press, 1982). 207, 209, 212, 213

18. K. Millington and F. J. Wright, 'Algebraic computations in elementary catastrophe theory', Proc. EUROCAL '85, European Conference on Computer Algebra, Linz, Austria, 1985, Lecture Notes in Comput. Sci. 204 (ed. B. F. Caviness, Springer, 1985) 116–125. 214, 216

19. D. M. Q. Mond, 'On the classification of germs of maps from $\mathbf{R}^2$ to $\mathbf{R}^3$', *Proc. London Math. Soc.* (3) 50 (1985) 333–369. 208

20. A. A. du Plessis, 'On the determinacy of smooth map-germs', *Invent. Math.* 58 (1980) 107–160. 208

21. T. Poston and I. N. Stewart, *Catastrophe theory and its applications* (Pitman, 1978). 212, 214

22. D. Ratcliffe, 'A classification of map-germs $\mathbf{C}^2, 0 \to \mathbf{C}^3, 0$ up to $\mathcal{A}$-equivalence', Preprint, University of Warwick, 1994. 214, 214

23. J. H. Rieger, 'Families of maps from the plane to the plane', *J. London Math. Soc.* (2) 36 (1987) 351–369. 208, 221

24. F. Tari, 'Recognition of $\mathcal{K}$-singularities of functions', *Experiment. Math.* 1 (1992) 225–229. 214

25. C. T. C. Wall, 'Finite determinacy of smooth map-germs', *Bull. London Math. Soc.* 13 (1981) 481–539. 207, 207, 208, 209, 212, 213, 213

26. J. M. West, 'The differential geometry of the crosscap', Ph.D. thesis, University of Liverpool, 1995. 208

N. P. Kirk `npkirk@hotmail.com` `http://www.liv.ac.uk/Maths/`

Department of Mathematical Sciences
The University of Liverpool
P.O. Box 147
Liverpool L69 3BX

https://doi.org/10.1112/S1461157000000280 Published online by Cambridge University Press