Manual for LIEPDE

Thomas Wolf
Department of Mathematics
Brock University
St.Catharines
Ontario, Canada L2S 3A1
twolf@brocku.ca

March 20, 2004

1 Purpose

The procedure `LIEPDE` computes infinitesimal symmetries for a given single/system of differential equation(s) (ODEs or PDEs)

$$H_A = 0. \tag{1}$$

To obey symmetries, differential equations (1) for unknown functions u^α of independent variables x^i must be form-invariant under infinitesimal transformations

$$\tilde{x}^i = x^i + \varepsilon\xi^i, \qquad \tilde{u}^\alpha = u^\alpha + \varepsilon\eta^\alpha \tag{2}$$

of first order in ε. To transform the equations (1) by (2), derivatives of u^α must be transformed, i.e. the part linear in ε must be determined. The corresponding formulas are (see e.g. [1], [2])

$$\begin{aligned} \tilde{u}^\alpha_{j_1\ldots j_k} &= u^\alpha_{j_1\ldots j_k} + \varepsilon\eta^\alpha_{j_1\ldots j_k} + O(\varepsilon^2) \\ \eta^\alpha_{j_1\ldots j_{k-1}j_k} &= \frac{D\eta^\alpha_{j_1\ldots j_{k-1}}}{Dx^k} - u^\alpha_{ij_1\ldots j_{k-1}}\frac{D\xi^i}{Dx^k} \end{aligned} \tag{3}$$

and the complete symmetry condition then takes the form

$$XH_A = 0 \qquad (\text{mod } H_A = 0) \tag{4}$$

$$X = \xi^i\frac{\partial}{\partial x^i} + \eta^\alpha\frac{\partial}{\partial u^\alpha} + \eta^\alpha_m\frac{\partial}{\partial u^\alpha_m} + \eta^\alpha_{mn}\frac{\partial}{\partial u^\alpha_{mn}} + \ldots + \eta^\alpha_{mn\ldots p}\frac{\partial}{\partial u^\alpha_{mn\ldots p}}, \tag{5}$$

where mod $H_A = 0$ means that the original PDE-system is used to replace some partial derivatives of u^α to reduce the number of independent variables, because the symmetry condition (4) must be fulfilled identically in x^i, u^α and all partial derivatives of u^α.

For point symmetries ξ^i, η^α are functions of x^j, u^β only. For more general higher order symmetries ξ^i, η^α may depend on derivatives of u^β. For those symmetries one can without loss of generality set $\xi^i = 0$ due to a symmetry of the symmetry conditions on the manifold of solutions of $H_A = 0$ themselves (e.g. §5.1 in [1]). The shifted generators

$$\tilde{\xi^i} = \xi^i + h^i, \quad \tilde{\eta^\alpha} = \eta^\alpha + h^i u^\alpha{}_{,i}$$

with arbitrary $h^i = h^i(x^j, u^\beta, \ldots, u^\beta_K)$ represent generators of the same symmetry.

2 Syntax of `LIEPDE`

The procedure `LIEPDE` is called through
`LIEPDE(`*problem,symtype,flist,inequ*`);`
All parameters are lists.

The first parameter specifies the DEs to be investigated:
problem has the form {*equations, ulist, xlist*} where

equations	is a list of equations, each has the form `df(ui,..)=...` where the LHS (left hand side) `df(ui,..)` is selected such that - The RHS (right h.s.) of an equations must not include the derivative on the LHS nor a derivative of it. - Neither the LHS nor any derivative of it of any equation may occur in any other equation. - Each of the unknown functions occurs on the LHS of exactly one equation.
ulist	is a list of function names, which can be chosen freely
xlist	is a list of variable names, which can be chosen freely

Equations can be given as a list of single differential expressions and then the program will try to bring them into the 'solved form' `df(ui,..)=...` automatically. If equations are given in the solved form then the above conditions are checked and execution is stopped it they are not satisfied. An easy way to get the equations in the desired form is to use

`FIRST SOLVE({`*eq1,eq2,...*`},{`*one highest derivative for each function u*`})`

(see the example of the Karpman equations in `LIEPDE.TST`). The example of the Burgers

equation in `LIEPDE.TST` demonstrates that the number of symmetries for a given maximal order of the infinitesimal generators depends on the derivative chosen for the LHS.

The second parameter *symtype* of `LIEPDE` is a list { } that specifies the symmetry to be calculated. *symtype* can have the following values and meanings:

`{"point"}` Point symmetries with $\xi^i = \xi^i(x^j, u^\beta),\ \eta^\alpha = \eta^\alpha(x^j, u^\beta)$ are determined.

`{"contact"}` Contact symmetries with $\xi^i = 0,\ \eta = \eta(x^j, u, u_k)$ are determined ($u_k = \partial u/\partial x^k$), which is only applicable if a single equation (1) with an order > 1 for a single function u is to be investigated. (The *symtype* `{"contact"}` is equivalent to `{"general",1}` (see below) apart from the additional checks done for `{"contact"}`.)

`{"general",`*order*`}` where *order* is an integer > 0. Generalized symmetries $\xi^i = 0$, $\eta^\alpha = \eta^\alpha(x^j, u^\beta, \ldots, u^\beta_K)$ of a specified order are determined (where $_K$ is a multiple index representing *order* many indices.)
NOTE: Characteristic functions of generalized symmetries ($= \eta^\alpha$ if $\xi^i = 0$) are equivalent if they are equal on the solution manifold. Therefore, all dependences of characteristic functions on the substituted derivatives and their derivatives are dropped. For example, if the heat equation is given as $u_t = u_{xx}$ (i.e. u_t is substituted by u_{xx}) then `{"general",2}` would not include characteristic functions depending on u_{tx} or u_{xxx}.
THEREFORE:
If you want to find *all* symmetries up to a given order then either
- avoid using $H_A = 0$ to substitute lower order derivatives by expressions involving higher derivatives, or
- increase the order specified in *symtype*.
For an illustration of this effect see the two symmetry determinations of the Burgers equation in the file `LIEPDE.TST`.

`{xi!_`*x1* `=...,...,`
`eta!_`*u1*`=...,...}` It is possible to specify an ansatz for the symmetry. Such an ansatz must specify all ξ^i for all independent variables and all η^α for all dependent variables in terms of differential expressions which may involve unknown functions/constants. The dependences of the unknown functions have to be declared

in advance by using the DEPEND command. For example,

```
DEPEND f, t, x, u$
```

specifies f to be a function of t, x, u. If one wants to have f as a function of derivatives of $u(t,x)$, say f depending on u_{txx}, then one *cannot* write

```
DEPEND f, df(u,t,x,2)$
```

but instead must write

```
DEPEND f, u!‘1!‘2!‘2$
```

assuming *xlist* has been specified as `{t,x}`. Because t is the first variable and x is the second variable in *xlist* and u is differentiated oncs wrt. t and twice wrt. x we therefore use `u!‘1!‘2!‘2`. The character `!` is the escape character to allow special characters like ‘ to occur in an identifier.

For generalized symmetries one usually sets all $\xi^i = 0$. Then the η^α are equal to the characteristic functions.

The third parameter *flist* of LIEPDE is a list { } that includes

- all parameters and functions in the equations which are to be determined such that symmetries exist (if any such parameters/functions are specified in *flist* then the symmetry conditions formulated in LIEPDE become non-linear conditions which may be much harder for CRACK to solve with many cases and subcases to be considered.)
- all unknown functions and constants in the ansatz `xi!_..` and `eta!_..` if that has been specified in *symtype*.

The fourth parameter *inequ* of LIEPDE is a list { } that includes all non-vanishing expressions which represent inequalities for the functions in flist.

The procedure LIEPDE returns a list containing a list of unsolved conditions if any, a list containing the general solution for ξ^i, η^α and a list of constants and functions appearing in the general solution or in the remaining unsolved conditions.

3 Flags, parameters

Two flags specify whether symmetry conditions are formulated and solved in stages or in one go.

If the equation to be investigated is of higher than first order and point symmetries are investigated then LIEPDE allows a set of preliminary conditions to be formulated and solved before formulating and solving the full set of conditions for this equation (more

details in [2], [3]). This successive execution is enabled by setting

```
LISP(PRELIM_:=t)$.
```

The default value is

```
LISP(PRELIM_:=NIL)$.
```

If the preliminary conditions are easy to solve completely then it is advantageous to formulate and solve them first, otherwise the formulation of the complete more overdetermined condition is better. Examples for both cases are given together with comments in `LIEPDE.TST`.

If symmetries of a system of equations are to be investigated then with the setting `LISP(INDIVIDUAL_:=t)$` conditions for the equations are formulated and solved individually which provides a speed up if symmetry conditions are very overdetermined. The default value is

```
LISP(INDIVIDUAL_:=NIL)$.
```

By default `LIEPDE` computes ξ and η for each symmetry. If a prolongation of the symmetry vector shall be calculated then the order of this prolongation can be specified by the setting `LISP(PROLONG_ORDER:= ...)$`.

Flags that control the solution of the symmetry conditions by `CRACK` are displayed with `CRACKHELP()$`. Among them are:

```
LISP (PRINT_:= NIL/0/1/ ...)$
```

`PRINT_=NIL` suppresses all CRACK output, for `PRINT_=n` (n a positive integer) `CRACK` prints only equations with at most n factors in their terms, and `OFF BATCH_MODE$` enables the interactive solution of the system of conditions with `CRACK`.

4 Requirements

`REDUCE 3.6` and the files `CRACK.RED, LIEPDE.RED` and all files `CR*.RED` which are read in from `CRACK.RED`.

```
IN "crack.red","liepde.red"$
```

(and appropriate paths) or compilation with

```
FASLOUT "crack"$
IN "crack.red"$
FASLEND$
FASLOUT "liepde"$
IN "liepde.red"$
FASLEND$
BYE$
```

and loading afterwards with `LOAD_PACKAGE crack,liepde$.`

References

[1] P.J. Olver, Applications of Lie Groups to Differential Equations, Springer-Verlag, New York (1986).

[2] H. Stephani, Differential Equations, Their solution using symmetries, Ed. M.A.H. MacCallum, Cambridge Univ. Press (1989).

[3] T. Wolf, An efficiency improved program LIEPDE for determining Lie-symmetries of PDEs, Proceedings of "Modern Group Analysis: advanced analytical and computational methods in mathematical physics", Acireale, Italy, October 1992, Kluwer Academic Publishers, pP 377-385, 1993.