pdynamics: A Maxima package for computation in Poisson dynamics

JA Vallejo[a,*,1]

[a]*Facultad de Ciencias, Universidad Autónoma de San Luis Potosí,*
Av Chapultepec 1570 Col. Lomas del Pedregal
CP 78295 San Luis Potosí (SLP) México

ARTICLE INFO

Keywords:
Poisson manifolds
Hamiltonian systems
Perturbation Theory
Normal Forms
Symbolic computations

ABSTRACT

A Maxima package called `pdynamics` is described. It is aimed to study Poisson (and symplectic) systems and, particularly, the determination of the second-order normal form for perturbed Hamiltonians $H_\epsilon = H_0 + \epsilon H_1 + \epsilon^2 H_2$, relative to the periodic flow of the unperturbed Hamiltonian H_0. The formalism presented here is global, it does not require recursive computations and allows an efficient symbolic implementation.

1. Introduction

In this paper we discuss some computational aspects of the normal form theory for Hamiltonian systems on general phase spaces, that is, Poisson manifolds. According to Deprit [8], a perturbed vector field

$$A = A_0 + \epsilon A_1 + \frac{\epsilon^2}{2} A_2 + \cdots + \frac{\epsilon^k}{k} A_k + O(\epsilon^{k+1})$$

on a manifold M, is said to be in normal form of order k relative to A_0 if $[A_0, A_i] = 0$ for $i \in \{1, \ldots, k\}$. In the context of perturbation theory, the normalization problem is formulated as follows: to find a (formal or smooth) transformation which brings a perturbed dynamical system to a normal form up to a given order.
The construction of a normalization transformation, in the framework of the Lie transform method [7, 11, 12, 14], is related to the solvability of a set of linear non homogeneous equations, called the homological equations. If the homological equations admit global solutions, defined on the whole M, we speak of a global normalization, which essentially depends on the properties of the unperturbed dynamics.

Here we are interested in the global normalization of a perturbed Hamiltonian dynamics relative to periodic Hamiltonian flows. In this case, a result due to Cushman [5], states that if A is Hamiltonian, and the flow of the unperturbed vector field A_0 is periodic, then the true dynamics admits a global Deprit normalization to arbitrary order. The corresponding normal forms can be determined by a recursive procedure (the so-called Deprit diagram) involving the resolution of the homological equations at each step.

In the paper [2], we extend Cushman's result to the Poisson case and derive an alternative coordinate-free representation for the second-order normal form, involving only three intrinsic operations: the averaging operators associated to the $\mathbb{S}^1-$action, and the Poisson bracket. We give a direct derivation of this representation based on a period-energy argument [10] for Hamiltonian systems, and some properties of the periodic averaging on manifolds [1, 5, 16]. This formalism allows us to get an efficient symbolic implementation for some models related to polynomial perturbations of the harmonic oscillator with 1:1 resonance. In particular, we compute the second-order normal form of the Hénon-Heiles [5], and the spring pendulum [3, 4, 9] Hamiltonians, expressed in terms of the Hopf variables.

The package `pdynamics`, written in the CAS Maxima [13], which can automatically compute the second-order normal form in most cases of interest. We have chosen this particular CAS because of its ease of use, its syntax (very similar to that used on a blackboard), and its open-source character. We give here a complete list of the functions contained in the package with examples of use for each one of them. The software can be downloaded from `https://github.com/josanvallejo/pdynamics`.

*Corresponding author
`jvallejo@fc.uaslp.mx` (J. Vallejo)
`http://galia.fc.uaslp.mx/~jvallejo` (J. Vallejo)
ORCID(s): `0000-0002-9508-1549` (J. Vallejo)
[1]Supported by a research grant from the Consejo Nacional de Ciencia y Tecnología (México) code A1-S-19428.

2. Installation

To use the package, simply copy the file `pdynamics.mac` in the `contrib` directory of your Maxima installation (in a Unix box this directory will be something as `/usr/share/maxima/5.42.0/share/contrib`, depending on the version number, and you will need root permissions for that, while in a Windows machine it will be located at `C:\Program Files\Maxima-5.42.0\share\maxima\5.42.0\share\contrib`). Another good option is to keep a copy of `pdynamics.mac` in your working directory (that is, the same directory where you are storing your current Maxima session). In both cases, it suffices to execute `load("pdynamics.mac")` in the Maxima prompt.

If you place the file somewhere else you can use the Maxima `file_search` command to tell the system where it is, as in `file_search("/home/johndoe/maxima/pdynamics.mac")`. In what follows, we assume that any one of these methods has been applied, so the package is available to Maxima.

```
(%i1)  load("pdynamics.mac")$
```

In this paper, we discuss the functions included in the package with examples of their usage. In the next section we offer an example of the code, the function `pbracket`, making some comments about its implementation, but due to space reasons we omit the code for the remaining functions. The interested reader can take a look at the source of the package, which for the most part is self-explanatory.

Let us say some words about the limitations of the package. As it is based on the computation of averages along the flow of a Hamiltonian vector field, it involves at some point the symbolic computation of an integral (when determining the integral curves that define the flow). Thus, the package is as good as it is the Maxima symbolic integrator, which is far from perfect. It may happen that some integral can be done 'by hand' and Maxima can not solve it, or that some other CAS (like Maple or Mathematica) can find the integral while Maxima can not. In these cases, the user can circumvent the difficulty by using this external output to directly define the Hamiltonian flow of the Hamiltonian under study, say h, as `phamflow(h):=[F1,...,Fn]`, where $F1$,...,Fn are the components of the flow, found by whatever means. But, of course, when some particular integral not solvable in closed form appears, the package is useless and numerical methods are required.

Another drawback is that the frequency function ω must be supplied by the user. Even in the simplest cases, to determine in an automated way whether a certain function is periodic and, if so, to compute its (shortest) period is a tricky task (for instance, consider the Dirichlet function $\mathcal{X}_{\mathbb{Q}}$), for Maxima or for any other CAS[2]. Thus, we have preferred to avoid it here. In most practical cases, the Hamiltonian will be a perturbation of the harmonic oscillator, so this seems to be not a serious problem[3].

3. Poisson brackets

Here we implement a function for computing the Poisson bracket of two functions $f, g \in \mathbb{R}^{2n}$:

$$\{f,g\} = \sum_{i=1}^{n} \left(\frac{\partial f}{\partial p_i}\frac{\partial g}{\partial q_i} - \frac{\partial f}{\partial q_i}\frac{\partial g}{\partial p_i} \right).$$

We must define first the functions in the form $f(q_1, p_1, ..., q_n, p_n)$. The order of the arguments is important, but their names are not. The function `pbracket` always returns the answer in the standard form (with coordinates $q_1, p_1, ..., q_n, p_n$).

[2] For a discussion related to Wolfram Alpha, see:
http://math.stackexchange.com/questions/141033/how-to-effectively-compute-a-periodic-function.

[3] Here is an example of what the user can do in most cases. Suppose we want to find the period of the function $f(t) = p_1 \sin \omega t/\omega + q_1 \cos \omega t$ (see 5.7 below). In a Maxima session, do `load(to_poly_solve)`, then `f(t):=p1*sin(w*t)/w+q1*cos(w*t)`, and `nicedummies(%solve(f(t+T)-f(t),T))`. Among the solutions, there is the one of interest for us: $2\pi k/\omega$, $k \in \mathbb{Z}$.

```
pbracket(f,g):=block([n,Q,P,vars],
n:length(args(lhs(apply(fundef,[f]))))/2,
Q:makelist(concat(q,j),j,1,n),
P:makelist(concat(p,j),j,1,n),
vars:join(Q,P),
lsum(
-diff(apply(f,vars),Q[i])*diff(apply(g,vars),P[i])
+diff(apply(f,vars),P[i])*diff(apply(g,vars),Q[i]),
i,makelist(j,j,1,n))
)$
```

The first lines of this function appear repeatedly in other functions, so let us briefly explain what they do. As the dimension n in $\mathbb{R}^{2n}$ is variable, the first thing to do is to know it, and this can be achieved by looking at the number of arguments of the functions f, g. The second line does that[4]. Thus, if f is of the form $f = f(a, b, c, d)$ we know that $n = 2$. Next, we form a list of the variables involved, naming them internally in a consistent way as $(q1, p1, ..., qn, pn)$. To work iteratively we need a labelling and we have chosen it to be (q_j, p_j) (note that given two lists $Q : [q1, ..., qn], P : [p1, ..., pn]$, Maxima's command `join` will intersperse them). This labelling is done independently of the user's one, so if the functions have been defined as, say, $f = f(a, b, c, d)$ and $g = g(x, y, z, t)$, we will treat them as functions $f = f(q1, p1, q2, p2), g = g(q1, p1, q2, p2)$ and the output $\{f, g\}$ will be again a function of $(q1, p1, q2, p2)$. The user can change then the name of the variables to whatever she wants.

3.1. Example

Consider in $\mathbb{R}^4$ with coordinates (q_1, p_1, q_2, p_2) (we denote $q = (q_1, q_2)$, $p = (p_1, p_2)$ and $|q| = \sqrt{q_1^2 + q_2^2}$, $|p| = \sqrt{p_1^2 + p_2^2}$) the functions $\frac{1}{2}|q|^2$, $\frac{1}{2}|p|^2$ and $q \cdot p = q_1 p_1 + q_2 p_2$:

```
(%i2)  normq(q1,p1,q2,p2):=(q1^2+q2^2)/2$

(%i3)  normp(q1,p1,q2,p2):=(p1^2+p2^2)/2$

(%i4)  prodqp(q1,p1,q2,p2):=q1*p1+q2*p2$
```

These functions close in $(C^\infty(\mathbb{R}^4), \{.,.\})$ a Lie subalgebra isomorphic to $\mathrm{sl}_2(\mathbb{R})$:

```
(%i5)  pbracket(normp,normq);

(%o5)  p2 q2 + p1 q1
(%i6)  pbracket(prodqp,normq);

(%o6)  q2^2 + q1^2
(%i7)  pbracket(prodqp,normp);

(%o7)  - p2^2 - p1^2
```

3.2. Example

Consider a central force in $\mathbb{R}^3$, F, with potential $V = U(r)$, where $r = \sqrt{x^2 + y^2 + z^2}$. The components of F are those of the gradient $-\mathrm{grad}V$, and they are computed using the chain rule, as the dependence of U with r is undetermined. We can work in Maxima with an arbitrary $U(r)$ declaring its (also undetermined) derivative to be $U'(r)$:

[4]Thanks to R. Dodier, from the Maxima list, for this trick.

```
(%i8)  V(x,y,z):=U(sqrt(x^2+y^2+z^2))$

(%i9)  gradef(U(r),U\'(r))$
```

For instance, $F_x = -\partial V/\partial x$ is given by

```
(%i10) diff(V(x,y,z),x);
```

$$(\%o10)\quad \frac{x\,U'\left(\sqrt{z^2+y^2+x^2}\right)}{\sqrt{z^2+y^2+x^2}}$$

For a particle moving in $\mathbb{R}^3$ under the influence of U, its Hamiltonian is

```
(%i11) Hcentral(x,px,y,py,z,pz):=(px^2+py^2+pz^2)/2 + V(x,y,z)$
```

and its angular momentum in the z direction

```
(%i12) L[z](x,px,y,py,z,pz):=x*py-y*px$
```

Then, no matter what the explicit expression of U is, the z component of the momentum is an integral of motion:

```
(%i13) pbracket(Hcentral,L[z]);
```

(%o13) 0

The same is true for the other components of $\mathbf{L} = (L_x, L_y, L_z)$[5].

4. Hamilton's equations

Given a Hamiltonian $H(q_1, p_1, ..., q_n, p_n)$, the evolution equation for any classical observable $f = f(q_1, p_1, ..., q_n, p_n)$ along a physical trajectory $c(t) = (q_1(t), p_1(t), ..., q_n(t), p_n(t))$ in phase space, is

$$\frac{\mathrm{d}c(t)}{\mathrm{d}t} = \{H, f\}. \tag{1}$$

In particular, the canonical (Hamilton's) evolution equations are:

$$\begin{aligned}\frac{\mathrm{d}q_i}{\mathrm{d}t} &= \frac{\partial H}{\partial p_i}\\ \frac{\mathrm{d}p_i}{\mathrm{d}t} &= -\frac{\partial H}{\partial q_i},\end{aligned} \tag{2}$$

for $i \in \{1, ..., n\}$. In order to write down the evolution equation for an observable, we first solve (2) and then substitute the obtained solutions $q_i = q_i(t), p_i = p_i(t)$ into (1).

[5] In fact, it is true that the square norm of the total angular momentum $L^2 = L_x^2 + L_y^2 + L_z^2$ has vanishing Poisson bracket with $Hcentral$, but we can't compute directly `pbracket(Hcentral,`$L_x^2 + L_y^2 + L_z^2$`)` because $L_x^2 + L_y^2 + L_z^2$ is not a single explicit function. While it is possible to implement properties such as bilinearity, the Jacobi identity or the Leibniz rule in the definition of `pbracket`, the form presented here is sufficient for our purposes.

4.1. Canonical equations

The following functions return and solve the canonical equations. The Hamiltonian must be defined first in the form $H = H(q_1, p_1, ..., q_n, p_n)$. The ordering of the arguments of H is important, but their names are not. The function `pcanonical_eqs` gives a list of the form $[eq_{q_1}, eq_{p_1}, ..., eq_{q_n}, eq_{p_n}]$ where each eq_{q_i}, eq_{p_i} is the first-order equation corresponding to $q_i(t)$, $p_i(t)$, respectively. On the other hand, `pcanonical_sol` returns a list of the form $[q_1(t), p_1(t), ..., q_n(t), p_n(t)]$, the solutions to these equations.

4.2. Example

The freely falling particle in an homogeneous gravitational field has a Hamiltonian:

```
(%i14) K(q,p):=p^2/2+m*g*q$
```

so the canonical equations are:

```
(%i15) pcanonical_eqs(K);
```

$$(\%o15)\quad [\frac{d}{d\,t}\,\mathrm{q1}\,(t) = \mathrm{p1}\,(t)\,,\frac{d}{d\,t}\,\mathrm{p1}\,(t) = -g\,m]$$

with solutions:

```
(%i16) pcanonical_sol(K);
```

$$(\%o16)\quad [\mathrm{q1}\,(t) = -\frac{g\,m\,t^2}{2} + \mathrm{p1}\,(0)\;t + \mathrm{q1}\,(0)\,,\mathrm{p1}\,(t) = \mathrm{p1}\,(0) - g\,m\,t]$$

4.3. Example

For the harmonic oscillator of frequency $\omega > 0$, with Hamiltonian:

```
(%i17) assume(%omega > 0);
```

$$(\%o17)\quad [\omega > 0]$$

```
(%i18) Hosc(x,y):=y^2/2+%omega^2*x^2/2;
```

$$(\%o18)\quad \mathrm{Hosc}\,(x, y) := \frac{y^2}{2} + \frac{\omega^2\,x^2}{2}$$

the canonical equations are:

```
(%i19) pcanonical_eqs(Hosc);
```

$$(\%o19)\quad [\frac{d}{d\,t}\,\mathrm{q1}\,(t) = \mathrm{p1}\,(t)\,,\frac{d}{d\,t}\,\mathrm{p1}\,(t) = -\omega^2\,\mathrm{q1}\,(t)]$$

with solutions:

```
(%i20) pcanonical_sol(Hosc);
```

$$(\%o20)\quad [\mathrm{q1}\,(t) = \frac{\mathrm{p1}\,(0)\,\sin\,(\omega\,t)}{\omega} + \mathrm{q1}\,(0)\,\cos\,(\omega\,t)\,,\mathrm{p1}\,(t) = \mathrm{p1}\,(0)\,\cos\,(\omega\,t) - \mathrm{q1}\,(0)\,\omega\,\sin\,(\omega\,t)]$$

4.4. Evolution of observables

The function `pevolution` computes the evolution of an observable $f = f(q_1, p_1, ..., q_n, p_n)$ with respect to a Hamiltonian $H = H(q_1, p_1, ..., q_n, p_n)$. It needs that the functions `pbracket` and `pcanonical_sol` be previously loaded.

4.5. Example

Let us do a little sanity check. If the observable is one of the coordinates q_i, its evolution must coincide with that resulting from the canonical equations. In this case we compute the evolution of the $q1$ coordinate of the harmonic oscillator (so the result should coincide with the first component in `(%o20)`):

```
(%i21) foo(a,b):=a$

(%i22) pevolution(foo,Hosc);
```

$$(\%o22)\quad \mathrm{F}(t) = \frac{\mathrm{p1}\,(0)\,\sin(\omega\,t)}{\omega} + \mathrm{q1}\,(0)\,\cos(\omega\,t)$$

5. Vector fields and flows

5.1. Integral curves of a vector field

The function `pintcurv` computes the integral curves of a vector field X in $\mathbb{R}^m$ (not necessarily even dimensional) given by the list of its components $[X_1, ..., X_m]$, each one of them a function of the local coordinates $[x_1, ..., x_m]$. The user must supply the list of components of X and a list containing the names of the coordinates.

5.2. Example

The integral curves of the vector field $X = (ax - b, ay)$ in the plane can be computed as follows:

```
(%i23) pintcurv([a*x-b,a*y],[x,y]);
```

$$(\%o23)\quad [\mathrm{x}\,(t) = \frac{b}{a} - \frac{(b - \mathrm{x}\,(0)\;a)\;e^{a\,t}}{a}, \mathrm{y}\,(t) = \mathrm{y}\,(0)\;e^{a\,t}]$$

5.3. The flow of a vector field

If X is a vector field on $\mathbb{R}^m$, its flow is a mapping $\mathrm{Fl}_X : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$ such that, if $c_p(t)$ is the integral curve passing by p at $t = 0$,

$$\mathrm{Fl}_X(t, p) = \mathrm{Fl}_X^t(p) = c_p(t).$$

The function `pvectflow` computes the flow of the vector field $X = (X_1, ..., X_m)$. As in the previous case, the input is a couple of lists: one containing the components of X, $[X_1, ..., X_m]$, and the other containing the coordinates used on $\mathbb{R}^m$, $[x_1, ..., x_m]$. It returns a list with the flow mapping components $[(\mathrm{Fl}_X)_1(t, x_1, ..., x_m), ..., (\mathrm{Fl}_X)_m(t, x_1, ..., x_m)]$.

5.4. Example

The flow of the vector field $X = (y, -\omega^2 x)$ in the plane is:

```
(%i24) pvectflow([y,-%omega^2*x],[x,y]);
```

$$(\%o24)\quad [\frac{\sin(\omega\,t)\;y}{\omega} + \cos(\omega\,t)\;x, \cos(\omega\,t)\;y - \omega\sin(\omega\,t)\;x]$$

5.5. The Hamiltonian vector field

Assume the manifold $\mathbb{R}^{2n}$ endowed with the canonical symplectic form

$$\Omega = \mathrm{d}p_1 \wedge \mathrm{d}q_1 + \cdots + \mathrm{d}p_n \wedge \mathrm{d}q_n. \tag{3}$$

Given a Hamiltonian $H = H(q_1, p_1, ..., q_n, p_n)$, its associated Hamiltonian vector field X_H is given by the condition $i_{X_H}\Omega = -\mathrm{d}H$. This is easily seen to lead to the components

$$X = \left(\frac{\partial H}{\partial p_1}, -\frac{\partial H}{\partial q_1}, ..., \frac{\partial H}{\partial p_n}, -\frac{\partial H}{\partial q_n}\right),$$

in the basis $\{\partial/\partial q_1, \partial/\partial p_1, ..., \partial/\partial q_n, \partial/\partial p_n\}$. The function `phamvect` computes X_H from H, expressing its components in the form $[X_1(q, p), ..., X_{2n}(q, p)]$.

5.6. Example

For the harmonic oscillator, the Hamiltonian vector field is:

```
(%i25) phamvect(Hosc);
```

(%o25) $[p1, -\omega^2\, q1]$

5.7. The Hamiltonian flow

Suppose we have a Hamiltonian vector field on $\mathbb{R}^{2n}$ endowed with the canonical symplectic form (3) above. Given a Hamiltonian $H = H(q_1, p_1, ..., q_n, p_n)$, the flow of its associated vector field X_H is called the Hamiltonian flow.
The function `phamflow` computes the Hamiltonian flow determined by a Hamiltonian H. It is just the composition of `pvectflow` and `phamvect`.
For example, if H is taken to be the Hamiltonian of the harmonic oscillator we recover previous results (cfr. Examples 5.4 and 5.6):

```
(%i26) phamflow(Hosc);
```

(%o26) $[\dfrac{p1\,\sin(\omega\, t)}{\omega} + q1\,\cos(\omega\, t)\,,\, p1\,\cos(\omega\, t) - \omega\, q1\,\sin(\omega\, t)]$

6. The averaging method for normal forms

6.1. Averaging of a function respect to a periodic flow

Suppose we have a Hamiltonian system (with phase space $\mathbb{R}^m$) on which there is an $\mathbb{S}^1-$action with generator X. Then, the flow Fl_X^t is periodic. The average of an observable g with respect to the induced $\mathbb{S}^1-$action is the function defined by

$$\langle g\rangle = \frac{1}{2\pi}\int_0^{2\pi} (\mathrm{Fl}_X^t)^* g\, \mathrm{d}t.$$

The `paverage` function below computes $\langle g\rangle$ given the input g, $X = [X_1, ..., X_m]$ (a list containing the components of the vector field X in the basis $\{\partial/\partial x_1, ..., \partial/\partial x_m\}$) and $x = [x_1, ..., x_m]$ (a list with the coordinates used in $\mathbb{R}^m$). The function g must have been previously defined.

Usually, the action is Hamiltonian, that is, the vector field $X = X_H$ for some Hamiltonian H. Assuming in this case that the phase space is $\mathbb{R}^{2n}$ with the canonical symplectic form (3), the function `phamaverage` computes the average of g with respect to the Hamiltonian vector field of H.

```
(%i27) goo(x,y):=-x^2*(1+y)/2$

(%i28) paverage(goo,[y,-x],[x,y]);
```

(%o28) $-\dfrac{y^2+x^2}{4}$

```
(%i29) Hosc0(q1,p1):=(q1^2+p1^2)/2$

(%i30) phamaverage(goo,Hosc0);
```

(%o30) $-\dfrac{q1^2+p1^2}{4}$

There is another average which is very important in the theory of normal forms. It is given by the action of the operator $\mathcal{S}$:

$$\mathcal{S}(g) = \frac{1}{2\pi}\int_0^{2\pi} (t-\pi)(\mathrm{Fl}_X^t)^* g\, \mathrm{d}t.$$

The command `psprojector` computes it:

```
(%i31) psprojector(goo,[y,-x],[x,y]);
```

$$(\%o31)\quad \frac{3\,x\,y-2\,x^3}{12}$$

6.2. Second-order normal form of a perturbed Hamiltonian

The previous routines are all we need for computing the normal form of a Hamiltonian on $\mathbb{R}^2$ endowed with the canonical symplectic form (3). If we have a system admitting an $\mathbb{S}^1-$action, described by a perturbed Hamiltonian

$$H = H_0 + \epsilon H_1 + \frac{\epsilon}{2} H_2,$$

and such that the Hamiltonian vector field of H_0, X_{H_0}, has periodic flow with frequency ω, then its second-order normal form is given by

$$H_0 + \epsilon\langle H_1\rangle + \frac{\epsilon^2}{2}\left(\langle H_2\rangle + \langle\{S(\frac{H_1}{\omega}), H_1\}\rangle\right).$$

The command `pnormal2` performs the necessary computations given the Hamiltonians H_0, H_1, H_2, and the parameter ϵ. Another variable ω (the frequency function for the flow of X_{H_0}) is optional: if it is not included, it is assumed that this frequency is $\omega = 1$. That function ω, if included in the argument of `pnormal2`, must have been previously defined.

6.3. Example: The Hénon-Heiles Hamiltonian

This example is taken from [6]. The Hamiltonian is

$$K = \frac{1}{2}(p_1^2 + p_2^2) + \frac{1}{2}(q_1^2 + q_2^2) + \epsilon\left(\frac{q_1^3}{3} - q_1 q_2^2\right)$$

(note that the perturbation term is an homogeneous polynomial of degree 3), so we define:

```
(%i32) K0(q1,p1,q2,p2):=(p1^2+p2^2)/2+(q1^2+q2^2)/2;
```

$$(\%o32)\quad \mathrm{K0}\,(q1,p1,q2,p2) := \frac{p1^2+p2^2}{2} + \frac{q1^2+q2^2}{2}$$

```
(%i33) K1(q1,p1,q2,p2):=q1^3/3-q1*q2^2;
```

$$(\%o33)\quad \mathrm{K1}\,(q1,p1,q2,p2) := \frac{q1^3}{3} - q1\,q2^2$$

```
(%i34) K2(q1,p1,q2,p2):=0;
```

$$(\%o34)\quad \mathrm{K2}\,(q1,p1,q2,p2) := 0$$

The frequency function for the flow of X_{K_0} is readily found to be (see footnote 3 in page 2):

```
(%i35)  u(q1,p1,q2,p2):=1$
```

The second-order normal form is then[6]:

```
(%i36) pnormal2(K0,K1,K2,%epsilon,u);
```

(%o36)

$$\frac{p2^2+p1^2}{2} + \frac{q2^2+q1^2}{2} - \frac{\epsilon^2}{48}\left(5\,q2^4 + \left(10\,q1^2 + 10\,p2^2 - 18\,p1^2\right)\,q2^2\right.$$
$$\left. +56\,p1\,p2\,q1\,q2 + 5\,q1^4 + \left(10\,p1^2 - 18\,p2^2\right)\,q1^2 + 5\,p2^4 + 10\,p1^2\,p2^2 + 5\,p1^4\right)$$

[6] We have slightly edited the output in order to make it more readable.

6.4. Hopf variables

It is usual to express the normal form in terms of the Hopf variables w_1, w_2, w_3, w_4, as a previous step to carry on the reduction of symmetry process (see [5],[6]). For the case in which H_0 is the Hamiltonian of the $2D-$harmonic oscillator, these variables form a system of functionally independent generators of the algebra of first integrals of H_0, and are defined as $w_1 = 2(q_1q_2 + p_1p_2)$, $w_2 = 2(q_1p_2 - q_2p_1)$, $w_3 = q_1^2 + p_1^2 - q_2^2 - p_2^2$, $w_4 = q_1^2 + q_2^2 + p_1^2 + p_2^2$. The functions `phopf2`, `phopf4` attempt to express a given expression (a homogeneous polynomial in the variables (q_1, q_2, p_1, p_2) of degree 2 and 4, respectively) in terms of them. To apply these functions to the output of `pnormal2` above, we can select the independent term and the coefficient of ϵ^2 as follows:

```
(%i37)  phopf2(coeff(%,%epsilon,0));
```

$$(\%o37)\quad \frac{w_4}{2}$$

```
(%i38) phopf4(coeff(%th(2),%epsilon^2));
```

$$(\%o38)\quad \frac{w_2^2\,(48\,\%r1+7)}{48} - \frac{w_4^2\,(48\,\%r1+5)}{48} + w_3^2\,\%r1 + w_1^2\,\%r1$$

The formulas appearing in [6] are recovered by choosing the value 0 of the parameter:

```
(%i39) subst(%r1=0,%);
```

$$(\%o39)\quad \frac{7\,w_2^2}{48} - \frac{5\,w_4^2}{48}$$

Thus, the second-order normal form of the Hénon-Heiles system is

$$H_\epsilon \circ \Phi_\epsilon = \frac{w_4}{2} + \frac{\epsilon^2}{48}\left(7w_2^2 - 5w_4^2\right) + O(\epsilon^3).$$

6.5. Example: The spring pendulum

Consider the case of the Hamiltonian of a spring-pendulum (see [4],[3],[9]):

$$H(q1,p1,q2,p2) = \frac{p_1^2+p_2^2}{2} + \frac{q_1^2+q_2^2}{2} - \frac{\epsilon}{2}q_1^2(1+q_2),$$

which is that of a perturbed system $H_0 + \epsilon H_1$, where

$$H_0(q1,p1,q2,p2) = \frac{p_1^2+p_2^2}{2} + \frac{q_1^2+q_2^2}{2},$$

and

$$H_1(q1,p1,q2,p2) = -\frac{q_1^2(1+q_2)}{2}.$$

Note that the perturbation term now is *not* homogeneous. We define the terms of the perturbed Hamiltonian:

```
(%i40) H0(q1,p1,q2,p2):=(p1^2+p2^2)/2+(q1^2+q2^2)/2;
```

$$(\%o40)\quad \mathrm{H0}\,(q1,p1,q2,p2) := \frac{p1^2+p2^2}{2} + \frac{q1^2+q2^2}{2}$$

```
(%i41) H1(q1,p1,q2,p2):=-q1^2*(1+q2)/2;
```

$$(\%o41)\quad \mathrm{H1}\,(q1,p1,q2,p2) := \frac{\left(-q1^2\right)\,(1+q2)}{2}$$

```
(%i42) H2(q1,p1,q2,p2):=0;
```

(%o42) $\mathrm{H2}\,(q1, p1, q2, p2) := 0$

and compute the normal form in the original variables[7]. Note that we do not explicitly write the frequency function (thus assuming it is the constant 1):

```
(%i43) pnormal2(H0,H1,H2,%epsilon);
```

(%o43)

$$\frac{p2^2+p1^2}{2}+\frac{q2^2+q1^2}{2}-\frac{\epsilon}{4}\left(q1^2+p1^2\right)$$
$$-\frac{\epsilon^2}{192}\left(\left(20\,q1^2-4\,p1^2\right)\,q2^2+48\,p1\,p2\,q1\,q2+5\,q1^4+\right.$$
$$\left.\left(-4\,p2^2+10\,p1^2+12\right)\,q1^2+20\,p1^2\,p2^2+5\,p1^4+12\,p1^2\right)$$

As before, we can express in terms of the Hopf variables the independent terms and the coefficient of ϵ:

```
(%i44)  phopf2(coeff(%,%epsilon,0));
```

(%o44) $\frac{w_4}{2}$

```
(%i45)  phopf2(coeff(%th(2),%epsilon,1));
```

(%o45) $-\frac{w_4}{8}-\frac{w_3}{8}$

Note that the coefficient of ϵ^2 is *not* a homogeneous polynomial (of degree 4): there are two 2−degree terms: $(q_1^2+p_1^2)/16$. Thus, it does not make sense to apply `phopf4`, as this would lead to an error. Luckily, these terms can be easily expressed in terms of the variables w_1, w_2, w_3, w_4 (as $(q_1^2+p_1^2)/16 = (w_4+w_3)/32$) and then we can analyse the remainder, which *is* a polynomial of degree 4:

```
(%i46) phopf4(coeff(%th(3),%epsilon,2)+12*(q1^2+p1^2)/192);
```

(%o46) $-\frac{w_4^2\,(768\,\%r2+25)}{768}+\frac{w_3^2\,(256\,\%r2+5)}{256}+\frac{w_2^2\,(32\,\%r2+1)}{32}+w_1^2\,\%r2-\frac{5\,w_3\,w_4}{384}$

Let us take the simplest solution:

```
(%i47) subst(%r2=0,%);
```

(%o47) $-\frac{25\,w_4^2}{768}-\frac{5\,w_3\,w_4}{384}+\frac{5\,w_3^2}{256}+\frac{w_2^2}{32}$

The remainder in the coefficient of ϵ^2 is

```
(%i48)  phopf2(12*(q1^2+p1^2)/192);
```

(%o48) $\frac{w_4}{32}+\frac{w_3}{32}$

Thus, we get the second-order normal form of the spring pendulum in the Hopf variables:

[7] Again, we have slightly edited the output.

$$H_\epsilon \circ \Phi_\epsilon = \frac{w_4}{2} - \frac{\epsilon}{8}\left(w_4 + w_3\right) + \frac{\epsilon^2}{32}\left(w_4 + w_3 + w_2^2 - \frac{25{w_4}^2}{24} - \frac{5w_3w_4}{12} + \frac{5{w_3}^2}{8}\right) + O(\epsilon^3).$$

7. Arbitrary resonances

Notice that the functions `phopf2` and `phopf4` assume that the Hopf variables are those corresponding to the 1:1 resonance, that is, the non-perturbed Hamiltonian is

$$H_0 = \frac{1}{2}(p_1 + w_1^2 q_1^2 + p_2 + w_2^2 q_2^2)\,,$$

with the particular values

$$w_1 = 1 = w_2\,.$$

To work in the case of an arbitrary resonance, the user must modify the definition of the associated Hopf variables. As we have seen, in the 1:1 resonance they are[8]

$$\begin{aligned}
\rho_1 =& 2(q_1q_2 + p_1p_2)\\
\rho_2 =& 2(q_1p_2 - q_2p_1)\\
\rho_3 =& q_1^2 + p_1^2 - q_2^2 - p_2^2\\
\rho_4 =& q_1^2 + q_2^2 + p_1^2 + p_2^2\,,
\end{aligned}$$

but in the case of the 1:2 resonance, for instance, they become

$$\begin{aligned}
\rho_1 =& q_1^2 + p_1^2\\
\rho_2 =& 4q_2^2 + p_2^2\\
\rho_3 =& p_2(p_1^2 - q_1^2) + 4p_1q_1q_2\\
\rho_4 =& 2q_2(p_1^2 - q_1^2) - 2q_1p_1p_2\,.
\end{aligned} \tag{4}$$

The best the user can do in these cases is to copy the corresponding function `phopfn` from the file `pdynamics.mac` and manually edit the definition of these quantities. The package contains an example, the function `phopf6`, which corresponds to the resonance 1:2 and expresses a polynomial of degree 6 in the (q_1, p_1, q_2, p_2) in terms of the Hopf variables (4) above.

References

[1] M. Avendaño-Camacho and Yu Vorobjev, 'Homological equations for tensor fields and periodic averaging', *Russian J. Math.Phys.*, **18** no. 3 (2011) 243–257.

[2] M. Avendaño-Camacho, J. A. Vallejo and Yu Vorobjev, 'A simple global representation for second-order normal forms of Hamiltonian systems relative to periodic flows'. J. of Phys. A: Math. and Theor. **46** (2013) 395201.

[3] E. Breitenberger and R. D. Mueller, 'The elastic pendulum: A nonlinear paradigm', *J. Math. Phys.*, **22** (1981) 1196–1211.

[4] R. Broucke and P. A. Baxa, 'Periodic solutions of a spring-pendulum system', *Celest. Mech.*, **8** 2 (1973), 261–267.

[5] R. Cushman, 'Normal forms for Hamiltonian vector fields with periodic flow', in *Geometric Methods in Mathematical Physics (S. Sternberg, ed.)* D. Reidel Publ. (1984) 125–144.

[6] R. Cushman, 'Geometry of perturbation theory', in *Deterministic Chaos in General Relativity. D. Hobill, A. Burd, A.A. Coley (eds.)* Nato Science Series B, Vol. 332, Springer Verlag (1993) 89–101.

[7] A. Deprit, 'Canonical transformation depending on a small parameter', *Celest. Mech.*, **1** no.1 (1969) 13–30.

[8] A. Deprit, 'Delaunay normalisations', *Celest. Mech.*, **26** no.1 (1982) 9–21.

[9] I. T. Georgiou, 'On the Global Geometric Structure of the Dynamics of the Elastic Pendulum', *Nonlinear Dynamics*, **18** Issue 1 (1999) 51–68.

[10] W. B. Gordon, 'On the relation between period and energy in periodic dynamical systems', *J. Math. Mech.*, **19** (1969) 111–114.

[11] G. Hori, 'Theory of general perturbations with unspecified canonical variables', *Publ. Astron. Soc. Japan*, **18** (1966) 287–296.

[12] A. A. Kamel, 'Perturbation method in the theory of nonlinear oscillations', *Celest. Mech.*, **3** (1970) 90–106.

[13] Maxima.sourceforge.net, *Maxima, a Computer Algebra System*. Version 5.28.0 (2012). `http://maxima.sourceforge.net/`.

[8] We change notation here to avoid confusion with frequencies, and denote the Hopf variables by the greek letter ρ.

[14] K. R. Meyer, 'Normal forms for Hamiltonian systems', *Celest. Mech.*, **9** (1974) 517–522.
[15] K. R. Meyer and G. R. Hall, *Introduction to Hamiltonian dynamical systems and the N-Body problem* (Springer-Verlag, New York, 1992).
[16] J. Moser, 'Regularization of Kepler's problem and the averaging method on a manifold', *Comm. on Pure and Appl. Math.*, **23**, Issue 4 (1970) 609–636.