A GAP package for computing with real semisimple Lie algebras
please use in H. Hong and C. Yap (Eds.): ICMS 2014, LNCS 8592, pp. 59 - 66. Springer, Heidelberg (2014).

Heiko Dietrich, Paolo Faccin, and Willem A. de Graaf

ABSTRACT. We report on the functionality and the underlying theory of the GAP package CoReLG (*Computing with Real Lie Groups*)[1]; it provides functionality to construct real semisimple Lie algebras, to check for isomorphisms, and to compute Cartan decompositions, Cartan subalgebras, and nilpotent orbits.
GAP, real semisimple Lie algebras

1. Introduction

An n-dimensional Lie algebra over a field $\mathbb{F}$ is an n-dimensional $\mathbb{F}$-vector space $\mathfrak{g}$, furnished with a bilinear multiplication

$$[-,-]\colon \mathfrak{g}\times\mathfrak{g}\to\mathfrak{g},\quad (a,b)\mapsto[a,b]$$

which satisfies $[u,u]=0$ and $[u,[v,w]]+[v,[w,u]]+[w,[u,v]]=0$ (Jacobi identity) for all $u,v,w\in\mathfrak{g}$. Studied originally over the complex field $\mathbb{F}=\mathbb{C}$, Lie theory originated in the 19-th century in the work of the Norwegian mathematician Sophus Lie. Since then it has developed tremendously and it has become one of the central areas of 20-th and 21-st century mathematics, finding many applications in such diverse fields as physics, geometry, and group theory. In the second half of the 20-th century, the development of the computer provided a new research tool in Lie theory. Algorithms were developed and implemented on computer for various tasks related to Lie theory. Initially, this mostly concerned the combinatorial formulae for investigating representations of Lie groups due to, for example, Weyl and Freudenthal. The success of this endeavour has led to a new field of research, called Computational Lie Theory, which is concerned with the development of algorithms in Lie theory, their implementation on computer, and their application to theoretical problems. Over the past decades several computer programs in this area have emerged, for example, LiE [3], GAP [8], and MAGMA [1]. The last two programs are large computer algebra systems having well developed libraries for Computational Lie Theory.

The main focus in Computational Lie Theory has been on complex semisimple Lie algebras and Lie groups, and their representations. However, an important branch of Lie theory deals with real Lie groups and algebras. These are of paramount importance in physics and differential geometry. Probably due to the difficulty with dealing with the field of real numbers, which leads to various phenomena of non-splitness, there has not been much attention to real Lie groups in Computational Lie Theory. This changed at the beginning of the 21-st century, when a large group in the United States set up a research program to study real Lie groups by computational means. This is known as the Atlas project [2], and its main goal is to study the unitary dual of a real Lie group. An important problem in real Lie theory,

Dietrich was supported by an ARC-DECRA Fellowship, project DE140100088.

not addressed by the Atlas project, is the classification of the orbits of a real Lie group acting on a vector space.

In this paper, we report on our GAP-package CoReLG [7], for working with real semisimple Lie algebras given by a multiplication table (which the Atlas software does not do). As described in detail in the book [9], defining a Lie algebra by its structure constants allows for a detailed investigation of its structure. We remark that efficient algorithms for dealing with complex semisimple Lie algebras are already available in the GAP-package SLA [10].

1.1. Notation. The aim of this section is to introduce necessary notation; we refer to any standard book (for example, [12], [13], and [14]) for details and proofs. The *structure constants* of an n-dimensional Lie algebra $\mathfrak{g}$ with basis $\{v_1, \ldots, v_n\}$ are $\{c_{a,b}^{(k)}\}_{1\leq a,b,k\leq n}$, defined by

$$[v_a, v_b] = \sum\nolimits_{k=1}^{n} c_{a,b}^{(k)} v_k.$$

The Lie algebra $\mathfrak{g}$ is *semisimple* if it has no nontrivial abelian ideals, or, equivalently, if it is the direct sum of *simple* Lie algebras, that is, nonabelian Lie algebras which have no nontrivial ideals. The *adjoint* of $g \in \mathfrak{g}$ is the map $\mathrm{ad}_{\mathfrak{g}}(g)\colon \mathfrak{g} \to \mathfrak{g}$, $h \mapsto [g, h]$. The *Killing form* of $\mathfrak{g}$ is the bilinear map $\kappa_{\mathfrak{g}}\colon \mathfrak{g} \times \mathfrak{g} \to \mathbb{C}$, $\kappa_{\mathfrak{g}}(g, h) = \mathrm{trace}(\mathrm{ad}_{\mathfrak{g}}(g) \circ \mathrm{ad}_{\mathfrak{g}}(h))$. Each semisimple Lie algebra $\mathfrak{g}$ defined over $\mathbb{C}$ has a *Cartan subalgebra* $\mathfrak{H} \leq \mathfrak{g}$, which is a maximal abelian subalgebra consists of semisimple elements, that is, each $h \in \mathfrak{H}$ has a diagonalisable adjoint. This gives rise to the *root space decomposition*

$$\mathfrak{g} = \mathfrak{H} \oplus \bigoplus\nolimits_{\alpha\in\Phi} \mathfrak{g}_\alpha \quad \text{where} \quad \mathfrak{g}_\alpha = \{g \in \mathfrak{g} \mid \forall h \in \mathfrak{H}\colon [h, g] = \alpha(h)g\};$$

where $\Phi \subseteq \mathfrak{H}^*$ consists of all linear maps $\mathfrak{H} \to \mathbb{C}$ such that $\mathfrak{g}_\alpha \neq \{0\}$. Since $\kappa_{\mathfrak{g}}$ is non-degenerate, for each $\alpha \in \Phi$ there exists $t_\alpha \in \mathfrak{H}$ with $\alpha(-) = \kappa(t_\alpha, -)$; for $\alpha, \beta \in \Phi$ define $(\alpha, \beta) = \kappa_{\mathfrak{g}}(t_\alpha, t_\beta) = \alpha(t_\beta)$. Now $V = \mathrm{Span}_{\mathbb{R}}(\Phi)$ is an Euclidean space with inner product $(-, -)$; this is a *root system*. The theory of abstract root systems shows that there exist a *basis of simple roots* $\Pi = \{\alpha_1, \ldots, \alpha_\ell\} \subseteq \Phi$ (which also is a vector space basis of V), and an associated Weyl group, Cartan matrix, and Dynkin diagram. The Cartan-Killing-Dynkin classification of simple complex Lie algebras states a one-to-one correspondence between the isomorphism types of these Lie algebras and the isomorphism types of Dynkin diagrams. The Dynkin diagrams are classified by their type: there are four infinite families A_n ($n \geq 1$), B_n ($n \geq 2$), C_n ($n \geq 3$), and D_n ($n \geq 4$), and five exceptional types G_2, F_4, E_6, E_7, and E_8.

If $\mathfrak{g}$ is a simple Lie algebra defined over the real numbers, then either $\mathfrak{g}$ is a simple complex Lie algebra considered as real, or the complexification $\mathfrak{g}^c = \mathfrak{g} \otimes_{\mathbb{R}} \mathbb{C}$ of $\mathfrak{g}$ is a simple complex Lie algebra. In the latter case, $\mathfrak{g}$ is a *real form* of the simple complex Lie algebra $\mathfrak{g}^c$. Each complex simple Lie algebra has, up to isomorphism, only finitely many real forms. A Cartan subalgebra of a real semisimple Lie algebra $\mathfrak{g}$ is a nilpotent self-normalising subalgebra $\mathfrak{h} \leq \mathfrak{g}$; its complexification $\mathfrak{h}^c$ is a Cartan subalgebra of $\mathfrak{g}^c$.

2. Applications

In this section, we describe some of the new functionality provided by our software package CoReLG [7]; we give details on the underlying theory in Section 3. Our algorithms and implementations allow to investigate real (semi)simple Lie algebras computationally: one can compute Cartan decompositions, Cartan subalgebras, nilpotent orbits, and isomorphisms between Lie algebras. The field `SqrtField` in the example output of our algorithms is the infinite-dimensional number field $\mathbb{Q}(\sqrt{-1}, \sqrt{2}, \sqrt{3}, \sqrt{5}, \sqrt{7}\ldots)$; we give more details in Section 4.

2.1. Construction of simple real Lie algebras. For every type of simple complex Lie algebra (A_n, B_n, C_n, D_n, G_2, F_4, E_6, E_7, E_8) there exist, up to isomorphism, only finitely many real forms. We provide functions `RealFormsInformation`, `IdRealForm`, and `RealFormById` which construct these real simple Lie algebras; the example below considers type A_3.

```
gap> RealFormsInformation("A",3);

  There are 5 simple real forms with complexification A3
    1 is of type su(4), compact form
    2 - 3 are of type su(p,4-p) with 1 <= p <= 2
    4 is of type sl(2,H)
    5 is of type sl(4,R)
  Index '0' returns the realification of A3

gap> L := RealFormById("A",3,4);
<Lie algebra of dimension 15 over SqrtField>
gap> IdRealForm(L);
[ "A", 3, 4 ]
```

2.2. Cartan subalgebras. We provide a function `CartanSubalgebrasOfRealForm` which constructs, up to conjugacy, all Cartan subalgebras of a real semisimple Lie algebra.

```
gap> L   := RealFormById("F",4,2);;
gap> CSA := CartanSubalgebrasOfRealForm(L);;
gap> Size(CSA);
8
gap> CSA[1];
<Lie algebra of dimension 4 over SqrtField>
```

2.3. Isomorphisms. The function `IsomorphismOfRealSemisimpleLieAlgebras` constructs, if exists, an isomorphism between two given real semisimple Lie algebras. The function `VoganDiagram` outputs the associated Vogan diagram, which determines the isomorphism type of the real form.

```
gap> L   := RealFormById( "F", 4, 2 );;
gap> sc  := StructureConstantsTable(Basis(L));;
gap> K   := LieAlgebraByStructureConstants(SqrtField,sc);;
gap> iso := IsomorphismOfRealSemisimpleLieAlgebras(K,L);
<Lie algebra isomorphism between Lie algebras of dimension 52>
gap> Display(VoganDiagram(L));
F4:  2---(4)=>=3---1
Involution: ()
```

2.4. Nilpotent orbits. The nilpotent orbits of a real simple Lie algebra $\mathfrak{g}$ are the G-orbits of nilpotent elements in $\mathfrak{g}$, where G is the adjoint group of $\mathfrak{g}$. If $\mathfrak{g}$ has rank at most 8, then the function `NilpotentOrbitsOfRealForm` computes representatives of the nilpotent orbits of $\mathfrak{g}$. These orbits have been precomputed and are stored in a database; they are constructed as orbits in $\mathfrak{g}$ by using the isomorphism functionality described above. The function `RealCayleyTriple` returns a so-called $\mathfrak{sl}_2$-triple defining the orbit; its third component is a nilpotent element representing the orbit.

```
gap> L:=RealFormById("A",3,3);;
gap> orb:=NilpotentOrbitsOfRealForm(L);;
gap> Length(orb);
9
gap> o:=orb[2];
<nilpotent orbit in Lie algebra>
gap> RealCayleyTriple(o);
```

```
[ (-1/4)*v.8+(-1/4)*v.14, (1/2)*v.2, (-1/4)*v.8+(1/4)*v.14 ]
```

Our algorithms can answer the following questions: Let $\mathfrak{g}$ be a real semisimple Lie algebra with semisimple subalgebra $\mathfrak{a} \leq \mathfrak{g}$; let $\mathfrak{s}$ be the semisimple part of the centraliser of $\mathfrak{a}$ in $\mathfrak{g}$; what is the structure of $\mathfrak{s}$, that is, its Cartan subalgebras, Cartan decompositions, and its isomorphism type? The following example considers the semisimple part $\mathfrak{c}$ of the centraliser of a subalgebra $\mathfrak{a} \leq \mathfrak{g}$ with $\mathfrak{a}$ and $\mathfrak{g}$ real forms of type A_1 and E_7, respectively.

```
gap> L:=RealFormById("E",7,2);;
gap> ch:=ChevalleyBasis(L);;
gap> A:=Subalgebra(L,[ch[1][1],ch[2][1],ch[3][1]],"basis");;
gap> C:=LieDerivedSubalgebra(LieCentraliser(L,A));;
gap> IdRealForm(C);
[ "D", 6, 5 ]
gap> Length(CartanSubalgebrasOfRealForm(C));
4
gap> Display(VoganDiagram(C));
                      (5)
                     /
D6:   1---2---3---4
                     \
                      6

Involution: ()
```

3. Underlying theory

We comment on the underlying theory for the tasks described in Section 2, see also [5] and [6].

3.1. Construction of simple real Lie algebras. The classification of the simple real Lie algebras is known, and, up to isomorphism, the real forms of a simple complex Lie algebra $\mathfrak{g}$ can be constructed as follows. The first step is straightforward and requires to construct the so-called *compact real form* $\mathfrak{c}$ of $\mathfrak{g}$. The associated *(compact) real structure* is $\tau\colon \mathfrak{g} \to \mathfrak{g}$, $a + \imath b \mapsto a - \imath b$, where $a, b \in \mathfrak{c}$; here we write $\mathfrak{g} = \mathfrak{c} \oplus \imath\mathfrak{c}$. Let $\theta \in \mathrm{Aut}(\mathfrak{g})$ be an automorphism of order 2, commuting with τ, and denote by $\mathfrak{c}_\pm$ the ± 1-eigenspace of the restriction of θ to $\mathfrak{c}$. Now $\mathfrak{r}_{\tau,\theta} = \mathfrak{c}_+ \oplus \imath\mathfrak{c}_-$ is a real form of $\mathfrak{g}$, and every real form of $\mathfrak{g}$ is isomorphic to $\mathfrak{r}_{\tau,\theta}$ for some θ. Moreover, $\mathfrak{r}_{\tau,\theta} \cong \mathfrak{r}_{\tau,\theta'}$ if and only if θ and θ' are conjugate in $\mathrm{Aut}(\mathfrak{g})$. Involutionary automorphisms of $\mathfrak{g}$ are classified, up to conjugacy, in terms of Vogan diagrams; running over these automorphisms yields all real forms of $\mathfrak{g}$ up to isomorphism.

Note that $\mathfrak{r}_{\tau,\theta} = \mathfrak{k} \oplus \mathfrak{p}$ where $\mathfrak{k}$ and $\mathfrak{p}$ are the 1- and (-1)-eigenspace, respectively, of the restriction of θ to $\mathfrak{r}_{\tau,\theta}$. This decomposition is a *Cartan decomposition* of $\mathfrak{r}_{\tau,\theta}$ with *Cartan involution* θ; it is unique up to conjugacy.

For a given simple complex Lie algebra and fixed Cartan subalgebra and *Chevalley basis*, there exist *canonical choices* for the compact real form and the involutionary automorphisms. The real forms, $\mathfrak{r}_{\tau,\theta}$, obtained from these choices are called real forms in canonical form. (We remark that the structure of the canonical automorphism is encoded in the Vogan diagram of the real form, see `VoganDiagram` above.)

The challenge to construct real forms efficiently is to determine the multiplication table of $\mathfrak{r}_{\tau,\theta}$ by theoretical means, which allows one to write down this table (and thus to define $\mathfrak{r}_{\tau,\theta}$) directly, avoiding all computations. This is a straightforward, but tedious undertaking; it requires to determine the structure constants of a suitable basis of each $\mathfrak{r}_{\tau,\theta}$.

3.2. Cartan subalgebras. If $\mathfrak{g}$ is a complex simple Lie algebra, then, up to conjugacy under its adjoint group, there is a unique Cartan subalgebra in $\mathfrak{g}$. In contrast, there is no unique Cartan subalgebra in a real simple Lie algebra. However, up to conjugacy, there are only finitely many Cartan subalgebras; they have been classified by Kostant (1955) and Sugiura (1959). For our implementation, we devised a constructive version of Sugiura's classification theorem; it depends on the notion of strongly orthogonal sets of roots.

3.3. Isomorphisms. Let $\mathfrak{g}$ be a simple real Lie algebra. By constructing and analysing its complexification $\mathfrak{g}^c$, we know the type of $\mathfrak{g}^c$. In particular, $\mathfrak{g}$ is isomorphic to some canonical real form $\mathfrak{r} = \mathfrak{r}_{\tau,\theta}$, with $\mathfrak{r}^c \cong \mathfrak{g}^c$, for some involutionary automorphism $\theta \in \mathrm{Aut}(\mathfrak{r}^c)$, commuting with the compact real structure τ of $\mathfrak{r}^c$. Our approach is to construct an isomorphism $\mathfrak{g}^c \to \mathfrak{r}^c$ which is compatible with the associated real structures of $\mathfrak{g}$ and $\mathfrak{r}$; such an isomorphism clearly induces an isomorphism between the real forms $\mathfrak{g}$ and $\mathfrak{r}$. We construct this isomorphism in several steps. First, for each Lie algebra $\mathfrak{g}$ and $\mathfrak{r}$, we construct a so-called maximally compact Cartan subalgebra and a Cartan involution stabilising this Cartan subalgebra. (Our implementations provide this functionality.) With respect to this Cartan subalgebra and chosen basis of simple roots, we construct a so-called Chevalley basis and canonical generating set; this allows us to define an explicit isomorphism $\varphi\colon \mathfrak{g}^c \to \mathfrak{r}^c$. The next step is to modify φ (by means of defining it with respect to a modified canonical generating set) so that φ is compatible with the Cartan involutions of $\mathfrak{g}$ and $\mathfrak{r}$, respectively. We achieve this by acting with the Weyl groups of $\mathfrak{g}^c$ and $\mathfrak{r}^c$ on the respective canonical generating sets. Once such a compatible φ is found, one can easily modify it again so that it is also compatible with the respective real structures. This completes the construction of an isomorphism $\mathfrak{g} \to \mathfrak{r}$.

The above construction assumes we know that $\mathfrak{g} \cong \mathfrak{r}$. In practice, we only work with $\mathfrak{g}$ and, using the above approach, find a suitable canonical generating set so that the Cartan involution acts on it in a *standard way*; in other words, we construct *standard parameters* for $\mathfrak{g}$, such that two real simple Lie algebras are isomorphic if and only if their standard parameters coincide. In this case, it is straightforward to write down an explicit isomorphism. By construction, the canonical forms $\mathfrak{r}_{\tau,\theta}$ have standard parameters.

3.4. Nilpotent orbits. The nilpotent orbits of a simple complex Lie algebra are determined by the Dynkin-Kostant and Bala-Carter classifications (see [4]): the nonzero nilpotent elements are in one-to-one correspondence to certain semisimple elements (*characteristic elements*), which are in one-to-one correspondence to certain weighted Dynkin diagrams. The situation is more complicated for a simple real Lie algebra $\mathfrak{g}$. By the Kostant-Sekiguchi correspondence, if $\mathfrak{g} = \mathfrak{k} \oplus \mathfrak{p}$ is a Cartan decomposition, then the nilpotent orbits in $\mathfrak{g}$ are in one-to-one correspondence to the nilpotent K-orbits in $\mathfrak{p}^c$, where K is the adjoint group of $\mathfrak{k}^c$. There exist efficient implementations for computing the K-orbits in $\mathfrak{p}^c$ (see [11]). However, the Kostant-Sekiguchi correspondence is non-constructive, and obtaining explicit orbit representatives in $\mathfrak{g}$ is difficult. We used ad hoc computations and Gröbner bases to make this correspondence explicit for Lie algebras of rank at most 8.

4. Technical problems

Our implementations face three technical (and theoretical) limitations.

Firstly, for a given real semisimple Lie algebra $\mathfrak{g}$, we have to construct a Cartan subalgebra of $\mathfrak{g}^c$ and a corresponding root system. While there exist efficient algorithms to construct Cartan subalgebras, computations of the associated root systems may fail because the algorithm does not succeed in splitting the Cartan subalgebra over a small-degree extension of the base field. The problem of finding Cartan subalgebras which can be split is very difficult.

Secondly, our current approach for making the Kostant-Sekiguchi correspondence explicit requires the use of ad hoc computations using Gröbner bases. Even though we automated these computations systematically, the complexity of Gröbner basis computations limits the scope of this approach. This, and some limitations of the algorithms in [11], are the reason why our databank of nilpotent orbits currently contains only Lie algebras of rank at most 8.

The final limitation is concerned with the base field. In order to define a Lie algebra by a multiplication table over the reals, it usually suffices to take a subfield of the real field as base field. However, many algorithms need a Chevalley basis which is defined over the complex numbers; therefore, we require that the base field also contains the imaginary unit $\imath$. Other procedures, for example, the isomorphism test, requires the computation of square roots. Thus, in practise, the base field of our real Lie algebras is $\mathbb{Q}^{\surd} = \mathbb{Q}(\imath, \sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots)$, the Gaussian rationals with all $\sqrt{p}$, p a prime, adjoined. We have implemented the arithmetic of this field in GAP, and realised it as the field `SqrtField`. We remark that, in theory, a computation with our implementation can fail because we cannot construct a particular square root; our observation is that this happens rather sporadically.

References

[1] W. Bosma, J. Cannon, and C. Playoust. *The Magma algebra system. I. The user language*, Computational algebra and number theory (London, 1993), J. Symbolic Comput. **24**, 235–265, 1997.

[2] F. du Cloux and M. van Leeuwen. *Software for structure and representations of real reductive groups*, v. 0.4.6, available from `www.liegroups.org`.

[3] A. M. Cohen, M. A. A. van Leeuwen, and B. Lisser. LiE *a Package for Lie Group Computations*, CAN, Amsterdam, 1992.

[4] D. H. Collingwood and W. M. McGovern. Nilpotent orbits in semisimple Lie algebras. Van Nostrand Reinhold Mathematics Series. Van Nostrand Reinhold Co., New York, 1993.

[5] H. Dietrich and W. A. de Graaf. *A computational approach to the Kostant-Sekiguchi correspondence,* Pacific J. Mathematics **265**, 349–379, 2013.

[6] H. Dietrich, P. Faccin, and W. A. de Graaf. *Computing with real Lie algebras: real forms, Cartan decompositions, and Cartan subalgebras,* J. Symbolic Comp. **56**, 27–45, 2013.

[7] H. Dietrich, P. Faccin, and W. A. de Graaf. `CoReLG` – computing with real Lie groups. A GAP4 package. `www.science.unitn.it/~corelg`, 2014.

[8] The GAP Group, GAP – Groups, Algorithms, and Programming, Version 4.5.5. `www.gap-system.org`, 2012.

[9] W. A. de Graaf. Lie Algebras: Theory and Algorithms. vol. 56 of North-Holland Math. Lib. Elsevier Science, 2000.

[10] W. A. de Graaf. `SLA` – computing with Simple Lie Algebras. A GAP4 package. `www.science.unitn.it/~egraaf/sla.html`, 2012.

[11] W. A. de Graaf. *Computing representatives of nilpotent orbits of θ-groups*, J. Symbolic Comput. **46**, 438–458, 2011.

[12] J. E. Humphreys. Introduction to Lie algebras and representation theory. Second printing, revised. Graduate Texts in Mathematics, 9. Springer-Verlag, New York-Berlin, 1978

[13] A. W. Knapp. Lie groups beyond an introduction. Second edition. Progress in Mathematics, 140. Birkhäuser, 2002.

[14] A. L. Onishchik. Lectures on Real Semisimple Lie Algebras and Their Representations. ESI Lectures in Mathematics and Physics. European Mathematical Society (EMS), Zürich, 2004.

SCHOOL OF MATHEMATICAL SCIENCES, MONASH UNIVERSITY, VIC 3800, AUSTRALIA
E-mail address: `heiko.dietrich@monash.edu`

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF TRENTO, POVO (TRENTO), ITALY
E-mail address: `faccin@science.unitn.it`

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF TRENTO, POVO (TRENTO), ITALY
E-mail address: `degraaf@science.unitn.it`

3 Lie Algebra Cohomology and the Theorems of Casselmanâ€"Osborne and. Kostant. 63.Â Let G be a complex semisimple Lie group and x a Borel subgroup B âŠ, G. Then the irre-ducible holomorphic representations of G lie in one-to-one correspondence with a certain subsetâ€"the set of so-called dominant weightsâ€"of the character group B. This correspon-dence identies an irreducible representation with its highest weight: If V is an irreducible representation.Â In some precise sense, the various Cartan subgroups of a real semisimple group G make contributions to the set of irreducible unitary representations. More specically, there are nitely many conjugacy classes of Cartan subgroups, and each conjugacy class makes a certain contribution. 4 On generating semisimple subalgebras of real simple Lie algebras. 5 Formality. 6 Tholozan's obstruction to compact Clifford-Klein forms in an algorithmic fashion.Â We create pairs (Lie algebra, semisimple subalgebra) using the database CoReLG [DFG1] and SLA [G]. 3 Preliminaries. Throughout this paper we use the basics of Lie theory without further explanations. One can consult [O1].Â These are dened as real forms of complex semisimple Lie algebras gC gen-erated by the Chevalley bases of gC over the reals. Let X be a Hausdor topological space and Î" a topological group acting on X. We say that an action of Î" on X is proper if for any compact subset S âŠ, X the set. {Î³ âˆˆ Î" | Î³(S) âˆ© S = âˆ…}. is compact. Computing with real Lie algebras: real forms, Cartan decompositions, and Cartan subalgebras. Heiko Dietrich, Paolo Faccin, and Willem A. de Graaf. J. Symb. Computation 56, 27 - 45 (2013). A GAP package for computing with real semisimple Lie algebras. Heiko Dietrich, Paolo Faccin, and Willem A. de Graaf. in H. Hong and C. Yap (Eds.): ICMS 2014, LNCS 8592, pp. 59 - 66. Springer (2014). Regular subalgebras and nilpotent orbits of real graded Lie algebras. Heiko Dietrich, Paolo Faccin, and Willem A. de Graaf. J. Algebra 423, 1044 - 1079 (2015). Nilpotent orbits in real symmetric pairs and stationa I know that it is mostly nothing but linear algebra computation, but it is annoying to do this manipulations by hand again and again everytime. I'll be greatly pleased if there is something. Thanks in advance for any help in this direction. computer-science lie-algebras math-software mathematica computer-algebra-systems. share | cite | improve this question |. TAX REFORMING USING A SOCIALLY ORIENTED MECHANISM Basnukaev M.Sh., Klyukovich Z.A., Sugarova I.V., Tuskaeva M.R., Bataev I.T. 259-264. 0. PLANNING AND FORECASTING IN THE PUBLIC TAX MANAGEMENT Basnukaev M.Sh., Sugarova I.V., Bataev I.T., Elzhurkaev I.Ya., Abdulazizova E.A. 265-269. 0.Â ETHNO-CULTURAL SPECIFICITY OF ENVIRONMENTAL ENERGY USE BY INDIGENOUS AND NOMADIC PEOPLES Bereznitsky S.V. 386-391. 0.Â JUSTIFICATION OF THE PRIORITIES OF REGIONAL DEVELOPMENT IN REAL SECTOR OF ECONOMY Idigova L.M., Gapaeva S.U., Mazhiev K.Kh., Mazhiev A.Kh., Dzhabrailova L.Kh., Plieva N.M. 1280-1286. 0.