University of Arizona

RTG Reseach Project

The Structure of the Socle and Radical Series for Projective Indecomposable Modules of Simple Groups

Author:
Rachel N. Baumann

Supervisor:
Klaus Lux

University of Arizona
Department of Mathematics

December 12, 2014

Contents

Chapter 1

Introduction

In this research project I computed experimental data in GAP on the socle and radical series of projective indecomposable modules of sfinite imple groups and I will discuss the structure of the series and their relations to each other. I will explain the methods used in GAP to construct the group algebra, projective indecomposable modules, and their socle and radical series. In closing I will look at a few examples and discuss some of the results.

1.1 Motivation

Let F be a field and let G be a finite group. There are many cases where group theoretic statements can be proved much easier using representation theory than group theory. A classic example of this is Burnside's $p^a a^b$ theorem:

Theorem 1.1. *If p and q are primes in $\mathbb{N}$ and G is a group of order $p^a q^b$ then G is solvable.*

The proof of this theorem by William Burnside is one of the best applications of representation theory since the proof is very short and straight forward whereas the proof using purely group theory is quite long and requires a much stronger backgroud in the theory of finite groups. Thus we see that there is a great benefit to using representation theory to prove group theoretic statements. Furthermore there is a very nice one-to-one correspondence between the F-representations of G and FG-submodules.My particular interest will be in looking at FG-modules for finite simple groups.

In studying modules we want to classifying all modules which leads to questions on the construction of modules. There are two approaches to constructing modules with one being more naïve than the other. But in order to see the benefits of one over the other I will explain both.

Let R be a ring with unity and let M be a module over R.

Definition 1.2. An R-module M is said to be *semisimple* if it is the direct sum of simple submodules.

Simple modules are the building block with which we construct modules. Thus the case of semisimple modules are easy to construct. However, many naturally occuring modules are not semisimple thus we need a different way to break up a module into simple pieces.

1.2 Constructing Modules

The first naïve way of constructing modules is by looking at the composition series of the module.

Definition 1.3. A *composition series* for M is a chain

$$0 < M_1 < M_2 < \cdots < M_{k-1} < M_k = M$$

of submodules orded by strict inclusion where M_i is a maximal submodule of M_{i+1}. The quotient modules M_{i+1}/M_i are called the *composition factors*.

Note that in general a composition series may not even exists. However, in the case that $R = FG$ for G a finite group, then R is a finite dimenaional algebra and hence R is both Artinian and Noetherian and thus it can be shown that any R-module M has a finite composition series. Furthermore the following theorem by Jordan and Hölder gives that the composition factors are uniquely determined upto permuation:

Theorem 1.4. *(Jordan-Hölder's Theorem)*
If

$$0 < M_1 < M_2 < \cdots < M_{k-1} < M_k = M$$

and

$$0 < N_1 < N_2 < \cdots < N_{t-1} < N_t = M$$

are both composition series of M then $t = k$ and to any composition factor M_{i+1}/M_i there is a composition factor N_{j+1}/N_j such that $M_{i+1}/M_i \simeq N_{j+1}/N_j$.

Now these composition factors of a series are simple R-modules giving us the simple pieces needed in constructing a module. However, even in having all the simple R-modules there are choices to be made in construcing the R-submodules in the composition series. This is not ideal since larger and larger modules will result in many more choices. This leads us to the second approach in constructing modules but we will need some more background in module theory.

Chapter 2

Some Module Theory

Now we look at the second approach to constructing modules by way of projective indecomposable modules of a group and looking at the quotients of direct sums of these modules. Let G be a finite group and F a field. We are interested in the case that the group algebra FG is not semisimple. Applying Maschke's Theorem on G gives us the cases of interest for the field characteristic of F.

Theorem 2.1. *(Maschke's Theorem) All FG-modules are semisimple if and only if the characteristic of F does not divide the order of G.*

Proof. Given by Carlson [1] on page 4. □

Thus the cases when $p \nmid |G|$ or when $\mathrm{char}(F) = 0$ are not interesting as these algebras are semisimple by Maschke's theorem and hence easily construcible from simple modules; so we're done in this case. Therefore, we restrict to the more interesting case when F is a field of characteristic p with $p \mid |G|$ and explore constructing FG-modules by looking at projective indecomposable modules as our simple pieces and measuring the closeness of these to being semisimple.

2.1 Projective Indecomposable Modules of a Group

Let R be a ring with unity, which in our specific case will be the group algebra FG, with G a finite simple group and F a field of characteristic p with $p \mid |G|$.

Definition 2.2. A R-module M is *indecomposable* if $M \neq 0$ and M cannot be written as a direct sum of two non-zero submodules.

A restatement of Maschke's Theorem in terms of simple (also called irreducible) and indecomposable modules gives the following relation:

Theorem 2.3. *(Maschke's Theorem) If* $\mathrm{char}(F) \nmid |G|$ *then irreducible (i.e simple) FG-modules are equivalent to indecomposable FG-modules.*

"Much of the role that simple modules play in complex representation theory is played in modular representation theory by indecomposable modules" ([2], 344). However, if $\mathrm{char}(F) \mid |G|$ then it is not true that irreducible and indecomposable are equivalent. An indecomposable module is weaker than a simple module. That is, any simple module is indecomposable but it is not the case that indecomposable modules are simle as seen in the following representation example.

Example 2.1. *Let $G = C_2 = \{1, x\}$ and $F = \mathbb{F}_2$. Consider the following representation*

$$\rho : G \to \mathrm{GL}_2(\mathbb{F}_2) \qquad \rho(x) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Now ρ is not irreducible as the subspace $U = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ is a ρ-invariant subspace. However, ρ is indecomposable. Indeed, U is the unique ρ-invariant subspace and hence we cannot decompose ρ. Therefore, we see that irreducibility is stronger than indecomposability.

Now these indecomposable modules are precisely the direct summands of FG and we'll see that there is a bijection between the isomorphism classes of projective indecomposable FG-modules and the isomorphism classes of simple FG-modules.

Theorem 2.4. *(Krull-Schmidt) Let M be a FG-module. Then M is a diect sum*

$$M = M_1 \oplus \cdots M_m$$

of indecomposable FG-modules $M_1, \ldots, M_m$. Moreover, if $M = M_1' \oplus \cdots \oplus M_{m'}'$ and $M_1', \ldots, M_{m'}'$ are indcomposable, then $m' = m$ and there is a permutation $\pi \in S_m$ such that $M_i \simeq M_{\pi(i)}'$ for $i = 1, \ldots m$.

For finite groups the number of isomorphic classes of simple modules is finite whereas the number of isomorphism classes of indecomposable modules is, in many cases, infinite. Thus we split the indecomposable modules into a subclass of projective indecomposable modules.

Definition 2.5. A R-module P is *projective* if to every epimorphism $f : M \to N$ of R-modules and to every homomorphism $g : P \to N$ there exists a homomorphism $h : P \to M$ with $f \circ h = g$.

$$\begin{tikzcd} & M \arrow[d, "f"] \\ P \arrow[ur, dashed, "\exists h"] \arrow[r, "g"'] \arrow[r, phantom, shift left=3, "\circlearrowright"] & N \end{tikzcd}$$

Example 2.2. *Every free module is a projective module. Indeed for F a free R-module and $\{e_i\}_{i\in I}$ a finite set of generators for F, that is $F = \bigoplus_{i\in I} Re_i$. If $f : M \to N$ is a surjective homomorphism of R-modules and $g : F \to N$ is a homomorphism then define $h : F \to M$ by $h(e_i) = f^{-1}(g(e_i))$.*

Definition 2.6. A *principal indecomposable module* (PIM) of a ring R is a submodule of R that is a direct summand of R and is indecomposable.

Definition 2.7. A *left semiperfect ring* R is a ring in which all left-modules X have projective covers (that is a pair (P, p) where P is a projective R-module and $p : P \to X$ a superfluous epimorphism.)

When over a semiperfect ring, as FG-modules are, projective indecomposable modules are principal indecomposable modules which I will refer to as PIMs. By example 2.2 every module is a quotient of a projective module. This leads to the correspondence between the isomorphism classes of simple R-modules and the isomorphism classes of projective indecomposable R-modules.

Theorem 2.8. *Let R be a semiperfect ring. Every simple R-module S has a unique (up to an isomorphism) indecomposable projective cover isomorphic to Re for some primitive idempotent $e \in R$. Every indecomposable projective module has a unique simple quotient (up to an isomorphism).*

Proof. See proof by Serganova [3]. □

Corollary 2.9. *Every indecomposable projective module over a semiperfect ring R is isomorphic to Re for some primitive idempotent $e \in R$. There is a bijection between the isomorphism classes of simple R-modules and isomorphism classes of projective indecomposable R-modules.*

Since the projective indecomposable FG-modules are precisely the direct summands of FG, by finding the PIMs and forming quotients of direct sums of PIMs we can construct a FG-module. So the goal is to find the PIMs of FG and formulate questions on the structure of these PIMs.

2.2 The Socle and Radical Series of PIMs

Let A be a finite dimensional algebra, specifically we will be taking $A = FG$, let b be a block of A (indecomposable two sided ideals of A) and $M \in b$ an indecomposable module.

Definition 2.10. The *socle* of an A-module M is the sum of all simple submodules of M,

$$\operatorname{soc}(M) = \sum \{N \mid N \text{ is a semisimple module of } M\}$$

Note that $\operatorname{soc}(M)$ is a maximal semisimple submodule of M. Dual to this is the radical of M which is the minimal submodule of M with semisimple quotient.

Definition 2.11. The *radical* of an A-module M is the intersection of all submodules N of M with M/N semisimple,

$$\operatorname{rad}(M) = \bigcap \{N \mid M/N \text{ is semisimple}\}$$

We will be looking at the case that M is a PIM and in particular the structure of the socle and radical series for a PIM which we define now.

Definition 2.12. The *socle series* (or *upper Loewy series*) for an indecomposable A-module M is defined by $s_1(M) = \operatorname{soc}(M)$ and

$$s_i(M)/s_{i-1}(M) = \operatorname{soc}(M/s_{i-1}(M)).$$

The socle series is a strictly increasing chain of submodules. If the socle series terminates in M then the minimal k such that $s_k(M) = M$ is called the socle length of M.

Definition 2.13. The *radical series* (or *lower Loewy series*) for an indecomposable A-module M is defined by $r_1(M) = \operatorname{rad}(M)$ and

$$r_i = \operatorname{rad}(r_{i-1}(M)),$$

where $M/\operatorname{rad}(M)$ is the *head* of M.

Dually, the radical series is a strictly decreasing chain of submodules. If the radical series terminates in 0 then the minimal k such that $r_k(M) = 0$ is called the radical length of M. Now, in general, the socle and radical series of a R-module M may not terminate. However, in the case of FG-modules M, since these are Artinian, we have that both series terminate; the socle series terminating in M and the radical series terminating in 0. Furthermore, the following theorem gives equality of the lenghts of these series.

Theorem 2.14. *Let R be an Artinian ring and let M be a R-module. Then the radical series of M terminates in 0 in the k^{th} step if and only if the socle series of M terminates in M in the k^{th} step.*

Proof. First as R is Artinian we have that both series terminate. Let l be the length of the socle series of a module M and let k be the length of the radical series of M. Then by defintion as $\operatorname{rad}_i(M)$ is the minimal submodule such that $\operatorname{rad}_{i+1}(M)/\operatorname{rad}_i(M)$ is semisimple, we have that

$$\operatorname{soc}_{l-i}(M) \supseteq \operatorname{rad}_i(M)$$

hence $l \geq k$. Also as $\operatorname{soc}_i(M)$ is the preimage of the maximal semisimple submodule of $M/\operatorname{soc}_{i-1}(M)$ we have that $l \leq k$ and hence $l = k$. □

This length $l(M)$ for both the socle and radical series is called the *Loewy Length.* Next, if we look at the socle and radical series of a PIM P then we have some nice results.

Theorem 2.15. *For P a projective indecomposable FG-module,* $\operatorname{soc}(P) \simeq P/\operatorname{rad}(P)$ *is a simple FG-module. Every simple FG-module is isomorphic to* $\operatorname{soc}(P) \simeq P/\operatorname{rad}(P)$ *for some PIM P.*

Therefore, for P a PIM we have that

$$\operatorname{soc}(p) = \operatorname{rad}_{l(P)-1}(P) \qquad \text{and} \qquad \operatorname{rad}(P) = \operatorname{soc}_{l(P)-1}(P).$$

Thus we might now ask, what happens with the intermediate terms of the series? When do we have equality of the socle and radical series for PIMs? If we don't have equality then are there bounds on how different the socle series can be from the radical series? That is can we bound the number of factors in the series which are different and or can we bound the number of differing simple modules between differing factors? These are questions we wish to explore and thus the reason for generating experimental data from simple groups so we can have some intuition in formulating theoretical conjectures for these questions.

Another interesting question to be asked is what can be said about the socle and radical series for projective indecomposable FG-modules as a collection? What about the subset with differing socle and radical series? One step in this direction is seen in Landrock's paper [4] where he gives conditions on the socle and radical series of the PIMs which in turn tell us that all PIMs have the same Loewy length.

Definition 2.16. A projective indecomposable FG-module P is said to be *upper-stable* (respectively *lower-stable*) if

$$\mathrm{rad}_2(P) = \mathrm{soc}_{l(P)-2}(P) \qquad (\text{respectively } \mathrm{soc}_2(P) = \mathrm{rad}_{l(P)-2}(P)).$$

Theorem 2.17. *Let S be a symmetric algebra, b a block of S. Assume that all projective indecomposable modules of b are either all upper-stable or all lower-stable. Then all projective indecomposable modules have the same Loewy length.*

Proof. See the proof in Landrock's paper [4] for theorem 1. □

We would like to explore this a bit further and see if anything can be said similarly when say the i^{th} and $(l(P)-i)^{th}$ factors of the socle and radical series are equal for all PIMs. Thus we now need to look at how to compute the socle and radical series of PIMs explicitly.

Chapter 3

Question on the Equality of the Socle and Radical Series

For this project I looked at FG-modules for simple groups working up from smaller simple groups to the larger ones. Now the projective indecomposable modules are rather complicated to work with which is why we make use of a computer. The main tool I used in this project was GAP, the "Groups, Algorithms, and Programming" software package designed for computational group theory [5].

3.1 Computational Group Theory Using GAP

In order to look at the strucuture of the socle and radical series of projective indecomposable FG-modules we need a way to compute the PIMs of FG. However, using FG and finding the PIMs is a naïve approach since not all simple FG-modules are known. Thus we take a different approach of looking at the basic algebra of FG.

Before I define the basic algebra of an algebra A we need some background. Denote $\mathfrak{Mod}(A)$ to be the category of of all right A-modules over a F-algebra A and $\mathfrak{mod}(A)$ to be the full subcategory of $\mathfrak{Mod}(A)$ whose objects are the finite dimensional right A-modules over F.

Definition 3.1. Two F-algebras A and B are *Morita equivalent* if there is an equivalence of the categories $\mathfrak{Mod}(A)$ and $\mathfrak{Mod}(B)$.

Proposition 3.2. *Let A and B be finite dimensional F-algebras and $\Psi : \mathfrak{Mod}(A) \to \mathfrak{Mod}(B)$ be a Morita equivalence. Then for every nonzero module M in $\mathfrak{mod}(A)$ the following hold:*

1. *M is a simple module in $\mathfrak{mod}(A)$ if and only if $\Psi(M)$ is a simple module in $\mathfrak{mod}(B)$.*

2. *M is an indecomposable module in $\mathfrak{mod}(A)$ if and only if $\Psi(M)$ is an indecomposable module in $\mathfrak{mod}(B)$.*

3. *M is a projective module in $\mathfrak{mod}(A)$ if and only if $\Psi(M)$ is a projective module in $\mathfrak{mod}(B)$.*

4. *Let $M_1, \dots M_n$ be nonzero modules in $\mathfrak{mod}(A)$. Then $M \simeq M_1 \oplus \cdots \oplus M_n$ in $\mathfrak{mod}(A)$ if and only if $\Psi(M) \simeq \Psi(M_1) \oplus \cdots \oplus \Psi(M_n)$ in $\mathfrak{mod}(B)$.*

For A a finite dimensional F-algebra with $e_1, \dots, e_r$ a complete set of primitive, orthogonal idempotents of A then the *basic algebra of* A, uniquely determined by A, is defined to be the algebra $A^b = eAe$ where $e = e_1 + \cdots + e_r$ [6]. Note that in practice finding idempotent elements of $A = FG$ can be tricky so we apply Maschke's trick of taking a subgroup $H \leq G$ with $\mathrm{char}(F) \nmid |H|$ and taking the idempotent element:

$$e_H = \frac{1}{|H|} \sum_{h \in H} h \in FG.$$

It's easy to check that $e_H FG e_H$ is a subalgebra of FG. This is a special idempotent called a fixidempotent of FG.

Theorem 3.3. *Every finite dimensional F-algebra A is Morita equivalent to its basic algebra A^b.*

Theorem 3.4. *Let A and B be finite dimensional F-algebras. Then A and B are Morita equivalent if and only if the basic algebra B^b and A^b.*

Proof. See proof by Skowroński and Yamagata ([6], page 174). □

Applying proposition 3.2 we have that two Morita equivalent algebras have the same number of simple modules as well as a bijection between their projective indecomposable modules. Also the structure of the socle and radical series are preserved under this equivalence. "An important class of finite dimensional algebras A are the indecomposable two sided ideals, also called blocks of a group algebra FG, where G is a finite group and F is a finite field" [7]. The GAP package `basic` is used to compute the basic algebra of a given algebra FG. In the current version of `basic` a fixidempotent of FG is used in constructing the basic algebra of FG. We then look at the PIMs of the basic algebra comupted by `basic` and compute the socle and radical series of these PIMs. A more through description and documentation of the package `basic` is given by Hoffman and Klaus [7].

3.2 Methods

This section gives an outline of what my GAP code does to compare the socle and radical series of projective indecomposable modules for G a simple group and F a finite field with characteristic dividing the order of G. Depending on the size of the group and the field characteristic, computing the basic algebra can be very time expensive. Thus for some of the larger simple groups I made use of the basic algebras already computed by Klaus Lux. The only difference in the two algorithms is in the input. One just takes in the name of the group and characteristic to compute the basic algebra using `basic` while the other takes in the already computed basic algebra. I will outline the later algorithm `srCheckAlg`:

1. Given a block B of the basic algebra FG^b computed using `basic`, run through all PIMs of the block B.
2. Construct the socle and radical series for each PIM P with respect to the standard database `db` of simple modules using the GAP functions `SocleSeries` and `RadicalSeries`.
3. Construct the matrix `mats` for the socle series of P, where the i^{th} row corresponds to $\mathrm{soc}_i(P)$ and each column corresponds to the simple modules of `db`. A nonzero entry (i,j) in the matrix represents the number of times the simple module corresponding to the j^{th} column appears in $\mathrm{soc}_i(P)$.
4. Construct the matrix `matr` for the radical series of P, where the i^{th} row corresponds to $\mathrm{rad}_{l(p)-i}(P)$ and each column corresponds to the simple modules of `db`. A nonzero entry (i,j) in the matrix represents the number of times the simple module corresponding to the j^{th} column appears in $\mathrm{rad}_{l(P)-i}(P)$.
5. A list of a single record `list[1]` is returned (in the future other characteristics for the group may be add to this list). The components of the list are as follows:
   - `groupName`: *list[1].groupName* is the name of the simple group G being investigated.
   - `characteristic`: *list[1].characteristic* is the characteristics of the basic algebra FG^b.
   - `srAllEqual`: *list[1].srAllEqual* is true if the socle and radical series of all PIMs in B are equal and false otherwise.
   - `PIMs`: *list[1].PIMs* is a list of records for each PIM of B. The components of this record are:

     `socleSeries`: *list[1].PIMs[i].socleSeries* is the matrix `mats` for PIM P_i.

     `radicalSeries`: *list[1].PIMs[i].radicalSeries* is the matrix `matr` for PIM P_i.

     `srEqual`: *list[1].PIMs[i].srEqual* is true if the socle and radical series for P_i are equal and false otherwise.

Note that `srCheck` is similar as above but includes the initial step of computing the basic algebra FG^b for a simple group G and field F. See the appendix for the source code for `srCheckAlg` and `srCheck`.

3.3 Example Computation For A_5

Here I demonstrate a basic example of how we compute in GAP the PIMs of FG^b and the socle and radical series of each PIM where $G = A_5$ and F is a field of characteristic 5. Using the AtlasRep and Basic package in GAP we begin by initializing our group algebra FG and by running AutoCalcBasic we compute the basic algebra FG^b considering the principal block (block 1), and construct the PIMs of this block. See documentation of `basic` package for specific details [7].

```
gap> alg:=InitializeRecordAtlasGroup("A5", 5, 1);;
gap> AutoCalcBasic(alg);
```

Using my function, `srCheckAlg`, we compute the matrices for the socle and radical series for each PIM of block 1 of FG^b, showing that there are two PIMs both of which have equal socle and radical series:

```
gap> Read("ReadSRseries");
#I  Have suborbits, compiling result record...
#I  Computing conjugate suborbits...
gap> srCheckAlg("A5", 5);
gap> list[1];
rec(
  PIMs := [
      rec( radicalSeries := [ [ 1, 0 ], [ 1, 1 ], [ 2, 1 ] ],
          socleSeries := [ [ 1, 0 ], [ 1, 1 ], [ 2, 1 ] ], srEqual := true ),
      rec( radicalSeries := [ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ],
          socleSeries := [ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ], srEqual := true ) ]
    , characteristic := 5, groupName := "A5", srAllEqual := true )
```

Walking through the code step by step, after we have constructed the PIMs for block 1 of FG and we compute the matrices for the socle and radical series for each PIM. Looking at just the second PIM, which I'll call P_2, first we create our standard database of simple FG^b-modules `db`, then we have the following socle and radical series in terms of this standard database of simple FG^b-modules, `db`:

```
gap> module := alg.PIMs[2].PIM;
<module of dim. 4 over GF(5)>
gap> db:=Chop(alg.global_simples[1]).db;
[ <trivial module of dim. 1 over GF(5)> ]
gap> for i in [2..Length(alg.global_simples)] do
> db:=Chop(alg.global_simples[i], rec(db:=db)).db;
> od;
gap> soc:=SocleSeries(module, db);
rec( basis := <cmat 4x4 over GF(5,1)>,
  cfposs := [ [ [ 1 ] ], [ [ 2 ], [ 3 ] ], [ [ 4 ] ] ],
  db := [ <trivial module of dim. 1 over GF(5)>,
      <abs. simple module of dim. 1 over GF(5)> ],
  ibasis := <cmat 4x4 over GF(5,1)>, isotypes := [ [ 2 ], [ 1, 2 ], [ 2 ] ],
  issoclerecord := true, module := <module of dim. 4 over GF(5)> )
gap> rad:=RadicalSeries(module, db);
rec( basis := <immutable cmat 4x4 over GF(5,1)>,
  cfposs := [ [ [ 1 ] ], [ [ 2 ], [ 3 ] ], [ [ 4 ] ] ],
  db := [ <trivial module of dim. 1 over GF(5)>,
      <abs. simple module of dim. 1 over GF(5)> ],
  ibasis := <immutable cmat 4x4 over GF(5,1)>,
  isotypes := [ [ 2 ], [ 1, 2 ], [ 2 ] ], isradicalrecord := true,
  module := <module of dim. 4 over GF(5)> )
```

From the list `isotypes` we see that there are three composition factors in both the socle and radical series and the values `[1]` and `[2]` correspond to the simple FG-modules in the standard database `db` we created:

```
gap> db;
[ <trivial module of dim. 1 over GF(5)>,
  <abs. simple module of dim. 1 over GF(5)> ]
```

Thus we see that indeed the factors in the both series match up since each corresponding factor consists of the same simple modules from `db`. In contrast to the output from `srCheck` for the second PIM we have the following 3×2 matrices:

```
gap> list[1].PIMs[2];
rec( radicalSeries := [ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ],
  socleSeries := [ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ], srEqual := true )
```

```
gap> list[1].PIMs[2].radicalSeries;
[ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ]
gap> list[1].PIMs[2].socleSeries;
[ [ 0, 1 ], [ 1, 2 ], [ 1, 3 ] ]
```

Where the number of rows correspond to the number of factors in the series of which we already know to be equal for both the socle and radical series and the number of columns corresponds to the number of simple module in `db`. Now for the matrix for the socle series, the first row represents the simple module in the first factor $\mathrm{soc}_1(P_2)$, thus as there is a 1 in the second column this means the factor is the simple module `db[2]`. Continuing, the second row is $\mathrm{soc}_2(P_2)$ which is a direct sum of the simple modules `db[1]` and two copies of `db[2]`. One nice feature of the matrix represenations of the socle series is that it is easy to see what the quotients $\mathrm{soc}_i(P_2)/\mathrm{soc}_{i-1}(P_2)$ are in terms of the simple FG-modules. For example $\mathrm{soc}_2(P_2)/\mathrm{soc}_1(P_2)$ is isomorphic to the direct sum of `db[1]` and `db[2]`. Next, the radical matrix is read from bottom to top. That is the first factor of the radical series is the last row of the matrix. In comparing both series factors, this is a good format as checking equality of rows corresponds to checking if $\mathrm{soc}_{l(P_2)-i}(P_2) = \mathrm{rad}_i(P_2)$. Indeed we see for both PIMs of A_5 in characteristic 5 that the socle and radical series are equal.

3.4 Example When the Socle and Radical Series Differ

In getting data on the socle and radical series of PIMs of simple groups one question that arose was for which groups and field characteristics do the series differ. It turned out that this is not as common and in order to find the right group and characteristic theoretically we need some background theory. The way we went about this was to look at module diagrams for constructing PIMs in the case that $p \mid |G|$ exactly once and in particular we looked at Brauer trees from a Brauer tree algebras to see that the Mathieu-11 group in characteristic 11 would give the desired result. First some definitions of Brauer tree and Brauer tree algebras.

Definition 3.5. (Alperin (as cited in [8])) A *Brauer graph* G is a finite connected graph, together with the following data:

(i) There exists a cyclic ordering of the edges adjacent to each vertex, usually described by the clockwise ordering given by a fixed planar representation of G;

(ii) For each vertex v, there exists a positive integer m_v assigned to v, called the *multiplicity*. We call a vertex v *exceptional* if $m_v > 1$.

Definition 3.6. A *Brauer tree* G is a Brauer graph which is a tree and having at most one exceptional vertex.
A *Brauer tree algebra* $A = A_G$ is a basic algebra given by a Brauer tree G as follows:

(i) There exists a one-to-one correspondence between simple A-modules S_i and edges i of G;

(ii) For any edge i of G, the projective indecomposable A-module P_i has $\operatorname{soc}(P_i) \simeq P_i/\operatorname{rad}(P_i)$ and $\operatorname{rad}(P_i)/\operatorname{soc}(P_i)$ is the direct sum of two uniserial modules [that is modules whose submodules are totally ordered by inclusion] whose composition factors are, for the cyclic ordering $(i, i_1, \ldots, i_a, i)$ of the edges adjacent to a vertex v, $S_{i_1}, \ldots, S_{i_a}, S_i, S_{i_1}, \ldots, S_{i_a}$ (from the top to the socle) where S_i appears $m_v - 1$ times.

Taking F to be a field of characteristic 11 and $G = M_{11}$ we note that $11 \mid |M_{11}|$ exactly once. As F-representations of G correspond to PIMs we look at the Brauer character table for M_{11} in characteristic 11 which leads to the following theorem:

```
M11mod11

      2  4  4  1  3  .  1  3  3
      3  2  1  2  .  .  1  .  .
      5  1  .  .  .  1  .  .  .
     11  1  .  .  .  .  .  .  .

        1a 2a 3a 4a 5a 6a 8a 8b
     2P 1a 1a 3a 2a 5a 3a 4a 4a
     3P 1a 2a 1a 4a 5a 2a 8a 8b
     5P 1a 2a 3a 4a 1a 6a 8b 8a
    11P 1a 2a 3a 4a 5a 6a 8a 8b

X.1      1  1  1  1  1  1  1  1
X.2      9  1  .  1 -1 -2 -1 -1
X.3     10 -2  1  .  .  1  A -A
X.4     10 -2  1  .  .  1 -A  A
X.5     11  3  2 -1  1  . -1 -1
X.6     16  . -2  .  1  .  .  .
X.7     44  4 -1  . -1  1  .  .
X.8     55 -1  1 -1  . -1  1  1

A = E(8)+E(8)^3
  = Sqrt(-2) = i2
```

Theorem 3.7. *In M_{11} the characters of degrees 11, 44 and 55 belond to blocks of defect 0 for characteristic 11. All other characters are in the principle block [9].*

By theorem 3.7 we have that there are five PIMs in the principal block and five simple modules corresponding to the characters of of degree 1, 9, 10, 10', and 16 which will be the five edges in the Brauer tree diagram $s_1, s_9, s_{10}, s_{16}, s_{10'}$ respectively. By applying (ii) of the definition of a Brauer tree algebra we have the following tree diagram for M_{11} in characteristic 11 below.

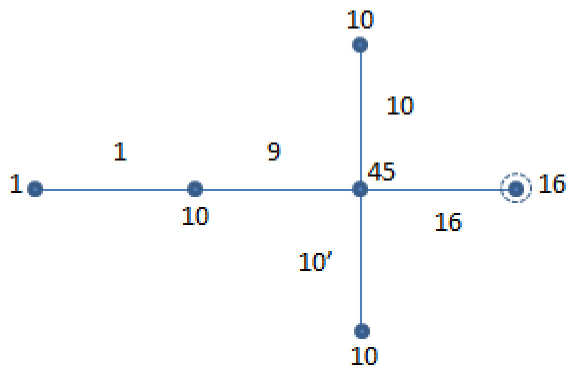

FIGURE 3.1: Brauer Tree for principal block of M_{11} in characteristic 11.

The edge labels are the Brauer character degrees corresponding to the simple modules and the vertices are labeled by sum of the character degrees on edges incoming to the vertex. The exceptional vertex is the one circled in figure 3.1. Now from the figure we see for the projective indecomposable FG-module P_9, as $\mathrm{rad}(P_9)/\,\mathrm{soc}(P_9)$ is a direct sum of two uniserial modules as described in definition 3.2, we have the following simple modules in the factors of the socle series of P_9 beginning with $\mathrm{soc}_1(P_9)$ and ending with $\mathrm{soc}_5(P_9)$:

$$\boxed{\begin{array}{c} \\ \\ \\ \\ s_9 \end{array}} \quad \boxed{\begin{array}{ccc} & & \\ & & \\ s_1 & \oplus & \\ & & s_{10'} \\ & s_9 & \end{array}} \quad \boxed{\begin{array}{ccc} & & \\ & & \\ s_1 & \oplus & s_{16} \\ & & s_{10'} \\ & s_9 & \end{array}} \quad \boxed{\begin{array}{ccc} & & \\ & & s_{10} \\ s_1 & \oplus & s_{16} \\ & & s_{10'} \\ & s_9 & \end{array}} \quad \boxed{\begin{array}{ccc} & s_9 & \\ & & s_{10} \\ s_1 & \oplus & s_{16} \\ & & s_{10'} \\ & s_9 & \end{array}}$$

Note that the last box is the body of the PIM P_9 and is obtained by listing the adjacent edges in the Brauer tree diagram to edge 9 in a counterclockwise fashion. Next we do the same for the radical series. Again by applying definition 3.2 we have the following simple modules in the factors of the radical series of P_9 beginning with $\mathrm{rad}_1(P_9)$ and ending with $\mathrm{rad}_5(P_9)$:

$$
\boxed{\begin{array}{ccc} & s_9 & \\ & & s_{10} \\ s_1 & \oplus & s_{16} \\ & & s_{10'} \\ & s_9 & \end{array}}
\quad
\boxed{\begin{array}{ccc} & & \\ & & s_{10} \\ s_1 & \oplus & s_{16} \\ & & s_{10'} \\ & s_9 & \end{array}}
\quad
\boxed{\begin{array}{cc} & \\ & \\ & s_{16} \\ & s_{10'} \\ s_9 & \end{array}}
\quad
\boxed{\begin{array}{cc} & \\ & \\ & \\ & s_{10'} \\ s_9 & \end{array}}
\quad
\boxed{\begin{array}{c} \\ \\ \\ \\ s_9 \end{array}}
$$

Note that we compare the reverse of the radical series with the socle series and thus see clearly that $\mathrm{soc}_i(P_9) \supseteq \mathrm{rad}_{6-i}(P_9)$ for all $1 \leq i \leq 5$. However, it's clear that $\mathrm{soc}_2(P_9) \neq \mathrm{rad}_4(P_9)$ and $\mathrm{soc}_3(P_9) \neq \mathrm{rad}_3(P_9)$ and thus we conclude that the socle and radical series for P_9 will not coincide. Note the expected matrices for the socle and radical series of P_9 with columns ordered based on the simple module order $s_1, s_9, s_{10}, s_{16}, s_{10'}$ are as follows:

$$
M_{\mathrm{soc}}(P_9) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 1 \end{bmatrix} \qquad M_{\mathrm{rad}}(P_9) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 1 \end{bmatrix}
$$

We note that only two factors are different and only by one simple module s_1. Next, checking in GAP by running `srCheck("M11", 11)`, the only PIMs with different radical and socle series are P_2 and P_5 in the list below; all the others have equal socle and radical series.

```
gap> list[1];
rec( PIMs :=[
      rec( radicalSeries := [ [ 1, 0, 0, 0, 0 ], [ 1, 1, 0, 0, 0 ],
              [ 2, 1, 0, 0, 0 ] ],
           socleSeries := [ [ 1, 0, 0, 0, 0 ], [ 1, 1, 0, 0, 0 ],
              [ 2, 1, 0, 0, 0 ] ], srEqual := true ),
      rec( radicalSeries := [ [ 0, 1, 0, 0, 0 ], [ 0, 1, 1, 0, 0 ],
              [ 0, 1, 1, 0, 1 ], [ 1, 1, 1, 1, 1 ], [ 1, 2, 1, 1, 1 ] ],
           socleSeries := [ [ 0, 1, 0, 0, 0 ], [ 1, 1, 1, 0, 0 ],
              [ 1, 1, 1, 0, 1 ], [ 1, 1, 1, 1, 1 ], [ 1, 2, 1, 1, 1 ] ],
          srEqual := false ),
      rec( radicalSeries := [ [ 0, 0, 1, 0, 0 ], [ 0, 0, 1, 0, 1 ],
              [ 0, 0, 1, 1, 1 ], [ 0, 1, 1, 1, 1 ], [ 0, 1, 2, 1, 1 ] ],
           socleSeries := [ [ 0, 0, 1, 0, 0 ], [ 0, 0, 1, 0, 1 ],
              [ 0, 0, 1, 1, 1 ], [ 0, 1, 1, 1, 1 ], [ 0, 1, 2, 1, 1 ] ],
          srEqual := true ),
```

```
      rec( radicalSeries := [ [ 0, 0, 0, 1, 0 ], [ 0, 1, 0, 1, 0 ],
              [ 0, 1, 1, 1, 0 ], [ 0, 1, 1, 1, 1 ], [ 0, 1, 1, 2, 1 ] ],
           socleSeries := [ [ 0, 0, 0, 1, 0 ], [ 0, 1, 0, 1, 0 ],
              [ 0, 1, 1, 1, 0 ], [ 0, 1, 1, 1, 1 ], [ 0, 1, 1, 2, 1 ] ],
          srEqual := true ),
      rec( radicalSeries := [ [ 0, 0, 0, 0, 1 ], [ 0, 0, 0, 1, 1 ],
              [ 0, 1, 0, 1, 1 ], [ 0, 1, 1, 1, 2 ], [ 0, 1, 1, 1, 3 ] ],
           socleSeries := [ [ 0, 0, 0, 0, 1 ], [ 0, 0, 0, 1, 2 ],
              [ 0, 1, 0, 1, 2 ], [ 0, 1, 1, 1, 2 ], [ 0, 1, 1, 1, 3 ] ],
          srEqual := false ) ], characteristic := 11, groupName := "M11",
  srAllEqual := false )
```

As the columns represent the simple modules in the database `db` we see that indeed, upto permuting the columns of P_2, the socle and radical matrices of P_2 are identical to those of P_9 computed by hand. Therefore, because of this example we do not have the case that the socle and radical series are equal for PIMs of simple groups. Some interesting observations from this example: (1) the lengths of the series for the PIMs with differing or the PIMs with equal socle and radical series are the same length and (2) the PIMs with differing series both differ by two factor and in each factor by one simple module.

3.5 Results

Over this semester I looked at about 100 different group algebras, FG. Of these about 30 had PIMs for which their socle and radical series differed. I have compiled the results from 27 of these groups in the tables below where only 17 are different simple groups.

TABLE 3.1: Data on groups whose socle and radical series for PIMs differ

Group	$Sz(8)$	$L_2(27)$	$Sz(8)$	$L_2(17)$	M_{11}	$L_2(8)$	M_{11}	$U_3(3)$	M_{11}	$U_3(3)$	$L_3(2)$	$U_3(3)$	A_7
Prime	2	7	5	3	11	3	2	2	3	3	2	7	2
Mult.	6	1	1	2	1	2	4	5	2	3	3	1	3
PIMs	7	2	4	2	5	2	3	3	7	8	3	3	3
DPs	7	1	1	1	4	1	1	2	6	8	2	1	2
TF	27	4	4	5	5	5	5	19	7	9	5	5	5
PSNTF	T	F	F	F	F	F	T	T	T	T	F	F	T
PMDF	16	1	1	2	2	2	2	14	2	6	2	2	2
PSNDF	T	T	T	T	T	T	T	F	T	F	T	T	T
PMDS	7	1	1	1	1	1	1	2	1	9	1	1	1
PSNDS	F	T	T	T	T	T	T	F	T	F	T	T	T

TABLE 3.2: Data on groups whose socle and radical series for PIMs differ

Group	$U_4(2)$	$L_2(31)$	M_{12}	$U_4(2)$	$L_2(23)$	$L_3(3)$	M_{12}	$L_3(3)$	A_8	$L_3(3)$	$L_2(32)$	$L_3(4)$	$L_2(13)$	$Sz(8)$
Prime	2	2	2	3	2	13	3	2	2	3	11	2	7	13
Mults.	6	5	6	4	3	1	3	4	6	3	1	6	1	1
PIMs	7	3	3	5	3	3	8	3	7	8	2	5	2	4
DPs	7	2	3	3	2	1	8	2	7	8	1	5	1	1
TF	13	17	10	13	5	5	9	5	13	9	6	13	4	5
PSNTF	T	F	T	F	F	F	T	T	T	T	F	T	F	T
PMDF	10	14	5	10	2	2	6	2	10	6	3	8	1	2
PSNDF	F	T	T	T	T	T	T	T	T	F	T	T	T	T
PMDS	25	1	6	24	1	1	12	1	19	9	1	24	1	1
PSNDS	F	T	F	F	T	T	F	T	F	F	T	F	T	T

Table acronyms

- Group: is the simple group used.
- Prime: is the characteristic of the basic algebra.
- Mult.: is the multiplicity of the prime in the factorization of the group order.
- PIMs: is the number of PIMs in the block B of the basic algebra.
- DPs : is the number of PIMs with different socle and radical series.
- TF: is the maximum number of factors in the socle series for all PIMs.
- PSNTF: is whether all PIMs have the same number of factors in the socle series.
- PMDF: is the maximum number of differing factors between the socle and radical series among all PIMs.
- PSNDF: is whether all differing PIMs differ by the same number of factors.
- PMDS: is the maximum number of differing simple modules between factors of the socle and radical series among all PIMs.
- PSNDS: is whether all differing factors differ by the same number of simple modules.

Observations

1. The prime 2 occurs most frequently in both tables. Is this a coincidence with this or something more happening in these cases? Also for the case $p = 2$, it does not seem like the socle and radical series are close to each other, that is the bounds on "how far off" they are should not be very good.
2. There are cases where the all of PIMs have differing socle and radical series. This seems to be related to having more differing factors and differing simples in those factors.

3. For many of the group algebras the number of differing factors is small compared with the total number of factors. Perhaps this should be the way of stating exactly what is meant by "how far off" the socle and radical series are from each other.

4. Also for many of the group algebras the number of differing simples is small. This should also be incorporated with defining what is meant by "how far off" the socle and radical series for a PIM are from each other. In particular the case $p \mid |G|$ exactly once seems to be a nice case. Can we say more exactly what "nice" means?

5. The groups that jump off as having drastically different socle and radical series are the following:

$$Sz(8)p2,\ U_3(3)p2,\ U_4(2)p2,\ M_{12}p2,\ U_4(2)p3,\ M_{12}p3,\ A_8p2,\ L_3(3)p3,\ L_3(4)p2$$

   Note that all but two are in characteristic 2. Does this still happen as we look at simple groups of larger and larger order?

3.6 Further Questions

There is still quite a bit of work to be done before some solid theoretical conjectures about the structure of the socle and radical series of projective indecomposable modules of simple groups can be made. However, this collection of data collected brings out some interesting questions that can now be further explored. Are there properties of the simple groups and field characteristics which cause the socle and radical series to be equal or differing? Right now I'm measuring "how far off" the socle series is from the radical series by looking at the differing factors and differing simple modules in those factors, but this should be made more rigorous and clear as to what is meant by a bound on how different these series can be. The case when $p \mid |G|$ exactly once seems very close to having equality of the series, can an exact bound be stated for the number of differing factors and the number of differing simple modules in those factors? What about the case when $p \mid |G|$ exactly twice or exactly n times? Is there a relation with the multiplicity of the prime dividing the group order and how different the series are? Also looking at the collection of PIMs as a whole, can statements similar to Landrock's theorem [4] be stated with perhaps weaker assumptions? Can anything be said when $\mathrm{soc}_i(P) = \mathrm{rad}_{l(P)-i}(P)$ and $\mathrm{rad}_i(P) = \mathrm{soc}_{l(P)-i}(P)$? Looking at the collection of PIMs with differing series when can we say they all have the same Loewy length or behave in the same way (that is same number of differing factors and same number of differing simple modules in those factors)? These are just a few of the questions yet to be throughly looked into more throughly. At present generating more data from different group algebras will be most helpful and hopefully in future work some of these questions can be answered.

Appendix A

Source Code

File GAPssrsCheck

```
# Gap code for checking the socle and radical series of a PIM of a group.
# Identify for which groups the socle and radical series coincide
# and for the ones that don't find the factors in the socle and radical
# that differ and by what simple modules they differ by.

Read("modatlasgroups");  # List of Simple Groups from the Altas Book.

# L2(25) is not in GAP fo some reason so need to remove it from the
# list modatlasgroups.
Remove(modatlasgroupsstring, 13);

g := "A5";;
alg := InitializeRecordAtlasGroup(g, 2,1);;
AutoCalcBasic(alg);;
module := alg.PIMs[1].PIM;;

# Initialize db using the first simple module in alg.global_simples

db := Chop(alg.global_simples[1]).db;;

# Look over the rest of the simple modules in alg.global_simples
# always updating the db calculated so far.
```

```
for m in [2..Length(alg.global_simples)] do
        db := Chop(alg.global_simples[m], rec(db:=db)).db;
od;

soc := SocleSeries(module, db);;
rad := RadicalSeries(module, db);;

list :=[];          # List for record of group.

listPIMs :=[];      # List of PIMs for group in specific characteristic.

mats :=[];          # Matrix for the socle series, keeping track of
                    # simple modules in each factor of the socle series.

matr :=[];          # Matrix for the radical series, keeping track of
                    # simple modules in each factor of the radical series.

i:=1;  j:=1;  k:=1;  n:=1;  p:=1;  q:=1;

# Record for each PIM of a group in a specific characteristic, keeps record
# of the PIM, the matrix for it's socle series and radical series, and true
# if the socle and radical series are equal (otherwise false).

a := rec( socleSeries := mats, radicalSeries := matr, srEqual := true);

# Record for each group in a specific characteristic, keeps record of the
# group, characteristic, list containing records are each PIM, and true if
# all socle and radical series are equal (otherwise false).

b := rec(groupName := "A5", characteristic := 2, PIMs := listPIMs,
          srAllEqual := true);

equal := true;
flag := true;



##################################################################################
```

```
##################################################################################

Socle Radical Equality Check for Simple Groups

Input - String name of simple group and a characteristic that divides the order
        of the group.

Output - List of record for group, containing group, characteristic, record
         for each PIM of the group, and true if all PIMs of the group have equal
         radical and socle series (false otherwise). Records for PIMs contain
         matrices for socle and radical series of PIM, and true if the radical
         and socle series are equal (false otherwise).

##################################################################################
##################################################################################


srCheck := function(group, char)
equal:= true;  flag:= true;
list:=[];  listPIMs:=[];

alg := InitializeRecordAtlasGroup(group, char, 1);
AutoCalcBasic(alg);

# get each PIM, with same db of simple modules across the board

for n in [1..Length(alg.PIMs)] do
         module := alg.PIMs[n].PIM;
         db := Chop(alg.global_simples[1]).db;

         for m in [2..Length(alg.global_simples)] do
                 db := Chop(alg.global_simples[m], rec(db := db)).db;
         od;

         # Create the Socle and Radical series for each PIM

         soc := SocleSeries(module, db);
         rad := RadicalSeries(module, db);
```

```
# Use matrices to keep track of the simples in each factor of the
# socle and radical series to compare.
# If the matrices are equal then the Socle and Radical series
# are the same, if not we can read off from the matrices by what
# factors they differ by in terms of simple modules.

mats := NullMat(Length(soc.isotypes), Length(alg.PIMs));
matr := NullMat(Length(rad.isotypes), Length(alg.PIMs));

# Create the socle series matrix which shows which simple modules
# are in each factor of the socle series.

for i in [1..Length(soc.isotypes)] do
        for k in [1..Length(soc.isotypes[i])] do
             for j in [i..Length(mats)] do
        mats[j][soc.isotypes[i][k]] :=
                            mats[j][soc.isotypes[i][k]]+1;
od;od;od;

# Create the radical series matrix which shows which simple modules
# are in each factor of the radical series. Start at the end of the
# radical series for easy comparision with the socle series.

for i in [1..Length(rad.isotypes)] do
        for k in [1..Length(rad.isotypes[Length(rad.isotypes)-i+1])] do
             for j in [i..Length(matr)] do
               matr[j][rad.isotypes[Length(rad.isotypes)-i+1][k]] :=
                matr[j][rad.isotypes[Length(rad.isotypes)-i+1][k]]+1;
od;od;od;

# Check if the socle and radical series for PIM are equal.

if mats = matr then equal := true;
else equal := false;
flag :=  false;
fi;

# Create PIM record and add to listPIMs.
a:= rec(socleSeries := mats, radicalSeries := matr, srEqual := equal);
```

```
         listPIMs[n] := a;
od;

# Create group record with listPIMs and add to list
b := rec(groupName := group, characteristic := char, PIMs := listPIMs,
          srAllEqual := flag);

Add(list, b);
end;
###################################################################################

                             End of Function srCheck

###################################################################################
```

File ReadSRseries

```
LoadPackage("basic");
Read("GAPssrsCheck");  # Code for Socle Radical Equality Check for simple groups
b := rec();

##################################################################################
##################################################################################

Socle Radical Equality Check for Algebra

Input - Group Algebra result of AutoCalcBasic and a characteristic that
        divides the order of the group.

Output - List of record for group, containing group, characteristic, record
         for each PIM of the group, and true if all PIMs of the group have
         equal radical and socle series (false otherwise). Records for PIMs
         contain matrices for socle and radical series of PIM, and true if
         the radical and socle series are equal (false otherwise).

##################################################################################
##################################################################################

srCheckAlg := function(alg)
equal := true;  flag := true;
list :=[];
listPIMs :=[];

# Get each PIM, with same db of simple modules across the board.

for n in [1..Length(alg.PIMs)] do
         module := alg.PIMs[n].PIM;
         db := Chop(alg.global_simples[1]).db;

         for m in [2..Length(alg.global_simples)] do
                 db := Chop(alg.global_simples[m], rec(db := db)).db;
         od;

         # Create the Socle and Radical series for each PIM.
```

```
soc := SocleSeries(module, db);
rad := RadicalSeries(module, db);

# Use matrices to keep track of the simples in each factor of the
# socle and radical series to compare.
# If the matrices are equal then the Socle and Radical series
# are the same, if not we can read off from the matrices by what
# factors they differ by in terms of simple modules from db.

mats := NullMat(Length(soc.isotypes), Length(alg.PIMs));
matr := NullMat(Length(rad.isotypes), Length(alg.PIMs));

# Create the socle series matrix which shows which simple modules
# are in each factor of the socle series.

for i in [1..Length(soc.isotypes)] do
 for k in [1..Length(soc.isotypes[i])] do
for j in [i..Length(mats)] do
        mats[j][soc.isotypes[i][k]] :=
                             mats[j][soc.isotypes[i][k]]+1;
od;od;od;

# Create the radical series matrix which shows which simple modules
# are in each factor of the radical series. Start at the end of the
   # radical series for easy comparision with the socle series.

for i in [1..Length(rad.isotypes)] do
        for k in [1..Length(rad.isotypes[Length(rad.isotypes)-i+1])] do
             for j in [i..Length(matr)] do
               matr[j][rad.isotypes[Length(rad.isotypes)-i+1][k]] :=
                matr[j][rad.isotypes[Length(rad.isotypes)-i+1][k]]+1;
od;od;od;

# Checks if the socle and radical series for PIM are equal

if mats = matr then equal := true;
else equal := false;
flag :=  false;
```

```
        fi;

        # Create PIM record and add to listPIMs.
        a := rec(socleSeries := mats, radicalSeries := matr, srEqual := equal);
        listPIMs[n] := a;
od;

# Create group record with listPIMs and add to list.
b := rec(groupName := alg.group, characteristic := alg.prime,
              PIMs := listPIMs, srAllEqual := flag);

Add(list, b);
end;
##################################################################################

                             End of Function srCheckAlg

##################################################################################
```

File landrockCrit

```
pims :=[];
truth := true;      truth2 := true;
loewyLength :=[];   n:=0;

##################################################################################
##################################################################################

Checking Theorem 1 Criterion from Landrock's Paper

Input - List outputed by the functions srCheck or srCheckAlg.

Output - Returns true if all PIMs are upper (lower) stable implies all PIMs
             have the same Loewy Length.

##################################################################################
##################################################################################

landrockCrit := function(list)

pims := list[1].PIMs;
truth := true;
loewyLength :=[];

for i in [1..Length(pims)] do

         # Check if each PIM is upper stable
         if pims[i].radicalSeries[1] = pims[i].socleSeries[1] then;
         else truth := false; fi;
od;

if truth = false then truth := true;
         for i in [i..Length(pims)] do

         # Check if each PIM is lower stable
         if pims[i].radicalSeries[Length(pims[i].radicalSeries)] =
                      pims[i].socleSeries[Length(pims[i].socleSeries)] then;
         else truth := false; fi;
```

```
            od; fi;

n:= Length(pims[1].radicalSeries);
loewyLength :=[];

for i in [1..Length(pims)] do
         Add(loewyLength,  Length(pims[i].radicalSeries), i);
         if Length(pims[i].radicalSeries) = n then;
         else truth2 := false; fi;
od;
return truth2;
end;
##################################################################################

                             End of Function landrockCrit

##################################################################################
```

Bibliography

[1] Jon F. Carlson and Ruedi Suter. *Modules and Group Algebras.* Birkhäuser Verlag, Basel, 1991.

[2] P.J. Hilton and U. Stammbach. *A Course in Homological Algebra.* Wiley, New York, 1998.

[3] Vera Serganova. Representation theory. URL (https://math.berkeley.edu/~serganov/math252/notes9.pdf).

[4] Peter Landrock. Some remarks on loewry lengths of projective modules. *Lecture Notes in Mathematics*, Volume 832:369–381, 1980.

[5] *GAP – Groups, Algorithms, and Programming, Version 4.7.6.* The GAP Group, 2014. URL (http://www.gap-system.org).

[6] Andrzej Skowronski and Kunio Yamagata. *Frobenius Algebras, Volume 1.* Zurich, Switzerland. European Mathematical Society, 2011.

[7] Tom Hoffman and Klaus Lux. *Basic Package Documentation*, June 2014.

[8] Takuma Aihara. Mutating brauer trees. Web. 6 Dec. 2014. URL (http://arxiv.org/pdf/1009.3210.pd).

[9] G.D. James. The modular characters of the mathieu groups. *Journal of Algebra*, Volume 27:57–111, 1973.