Computations of the Adams-Novikov E_2-term

Guozhen Wang

April 3, 2020

Abstract

1 Introduction

The Adams-Novikov spectral sequence is an important tool for the study of stable homotpy groups. One of the difficulty for this method is that the computations of the E_2 term is very hard. This paper gives an efficient algorithm for computing the Adams-Novikov E_2 term using computers.

Usually people compute the Adams-Novikov E_2 by the following three methods: the algebraic Novikov spectral sequence, the Bockstein spectral sequence, and the chromatic spectral sequence. The chromatic spectral sequence separates informations of different heights and compute them individually. This gives many global structure theorems for the Adams-Novikov spectral sequence. However, for stem-wise computations, especially for small primes, it is hard to get complete informations using the chromatic methods alone. For example, to get to the stem 126 at prime 2, informations up to chromatic level 6 are involved, which are beyond our currently knowledge.

The Bockstein spectral sequence is often used for stem-wise computations. For example, this method is illustrated in [3]. The method used in [3] to compute the differentials in the Bockstein spectral sequence is to compute in the cobar complex. As the cobar complex grows very fast, direct computations in the cobar complex becomes impossible very quickly, even for computers.

The algebraic Novikov spectral sequence is also often used in stem-wise computations. This spectral sequence also is very useful for comparisons with other spectral sequence. For example, it is proved in [2] that the algebraic Novikov spectral sequence for the sphere is isomorphic to the motivic Adams spectral sequence for the cofiber of tau. As is in the case of the Bockstein spectral sequence, the method of computing the algebraic Novikov differentials using the cobar complex becomes too complicated very quickly.

The main result of this paper is the observation that, the Bockstein and the algebraic Novikov filtrations are defined on any resolution of BP_*BP-comodules. So we can replace the cobar complex with any cofree resolution. In particular we introduce the notion of a minimal resolution which is smallest among cofree resolutions.

We define a cofree resolution to be minimal, if its mod I reduction is a minimal resolution as BP_*BP/I-comodules. Then we use the Bockstein and the algebraic Novikov filtrations on the minimal resolution to compute the Bockstein and the algebraic Novikov spectral sequences respectively.

To construct the minimal resolution, we first construct a minimal resolution for the mod I reduction. Then an arbitrary lift of this resolution almost gives us what we want, except that the compositions of consecutive maps are only zero modulo I. Then we do adjustments using some kind of Gaussian elimination. One such algorithm is given in Section 4. The naive method can be optimized by observing that, for maps between cofree comodules, we only need to know the projection to the cogenerators. This reduces the size of the matrices in the resolution by one order. This optimization is given in Section 5.

2 Notation

Let BP be the Brown-Peterson spectrum. It is a complex oriented ring spectrum whose associated formal group law is the universal p-typical formal group law over $\mathbb{Z}_p$. We have

$$BP_* = \mathbb{Z}_p[v_1, v_2, \dots]$$
$$BP_*BP = BP_*[t_1, t_2, \dots].$$

Define

$$I = (p, v_1, v_2, \dots)$$
$$P = BP_*BP/I = \mathbb{F}_p[t_1, t_2, \dots].$$

The pair (BP_*, BP_*BP) forms a Hopf algebroid which represents the moduli stack of formal groups over $\mathbb{Z}_p$.

Reduction modulo I gives the Hopf algebra P, which is a sub-Hopf algebra of the dual Steenrod algebra. For $p = 2$, P is isomorphic to the dual Steenrod algebra with degrees doubled.

A BP_*BP-comodule F is **cofree** if F is a direct product of copies of BP_*BP (and its degree shifts).

For a BP_*BP-comodule M, let $\mathrm{Prim}(M)$ denote the primitive elements of M.

We carry out all constructions in the graded sense. In particular, a direct product of objects which are finite in each degree is also a direct sum.

Definition 1. *A graded $\mathbb{Z}_p$-module is locally finite if it is bounded below and finitely generated in each degree.*

For example, BP_*, BP_*BP, and P are all locally finite. We implicitly assume that all modules are locally finite.

3 Minimal resolutions of BP_*BP-comodules

In this section we introduce the notion of a minimal resolution for BP_*BP-comodules, which are lifts of minimal resolutions of P-comodules.

Let M be a BP_*BP-comodule which is free over BP_*. Then M/I is a P-comodule.

Definition 2. *A comodule map is a strong injection (resp., strong surjection) if it is a split injection (resp., split surjection) of underlying BP_*-modules.*

An exact sequence

$$0 \to M \xrightarrow{f} F \xrightarrow{g} N \to 0$$

of BP_*BP-comodules is strongly exact if f is strongly injective and g is strongly surjective.

Remark 3.1. *A map is strongly injective (resp., surjective), if its associated matrix over BP_* can be transformed into the form $\begin{pmatrix} id \\ 0 \end{pmatrix}$ (resp., $\begin{pmatrix} id & 0 \end{pmatrix}$) by row (resp., column) transformations.*

For a comodule map

$$f : M \to N,$$

let $\tilde{f}$ be the reduction $M/I \to N/I$ of f modulo I. If f is strongly injective (resp., strongly surjective), the cokernel (resp., kernel) of $\tilde{f}$ is the reduction of $\operatorname{coker}(f)$ (resp., $\ker(f)$) modulo I.

Proposition 3. *Let M and N be BP_*BP-comodules which are locally finite and free as BP_*-modules. Then a comodule map*

$$f : M \to N$$

is strongly injective (resp., strongly surjective) if and only if the reduction

$$\tilde{f} : M/I \to N/I$$

modulo I is injective (resp., surjective).

Proof. If $\tilde{f}$ is injective, then by lifting the row and column transformations, the matrix for f can be transformed into one which is equivalent to $\begin{pmatrix} id \\ 0 \end{pmatrix}$ modulo I. Since I is a maximal ideal in BP_*, one can further transform the matrix into $\begin{pmatrix} id \\ 0 \end{pmatrix}$. □

Corollary 4. *Suppose we have a sequence*

$$0 \to M \xrightarrow{f} F \xrightarrow{g} N \to 0$$

of BP_*BP-comodule maps such that M, F, and N are free over BP_*, and $g \circ f = 0$. The sequence is strongly exact if and only if its reduction modulo I is exact.

Recall that the data of a long exact sequence

$$0 \to M \to F_0 \to F_1 \to \cdots$$

is equivalent to the data of a sequence

$$0 \to M \to F_0 \to M_1 \to 0$$

$$0 \to M_1 \to F_1 \to M_2 \to 0$$

$$0 \to M_2 \to F_2 \to M_3 \to 0$$

$$\vdots$$

of short exact sequences.

Definition 5. *A long exact sequence*

$$0 \to M \to F_0 \to F_1 \to \cdots$$

of BP_*BP-comodules is a cofree resolution of M if each F_i is cofree, and each short exact sequence

$$0 \to M_i \to F_i \to M_{i+1} \to 0$$

is strongly exact.

Recall we have the following notion of minimal resolutions of P-comodules.

Definition 6. *A cofree resolution*

$$0 \to \tilde{M} \to \tilde{F}_0 \to \tilde{M}_1 \to 0$$

$$0 \to \tilde{M}_1 \to \tilde{F}_1 \to \tilde{M}_2 \to 0$$

$$0 \to \tilde{M}_2 \to \tilde{F}_2 \to \tilde{M}_3 \to 0$$

$$\vdots$$

of a P-comodule $\tilde{M}$ is called minimal if for all $i \geq 0$, the induced map

$$\mathrm{Prim}(\tilde{M}_i) \to \mathrm{Prim}(\tilde{F}_i)$$

is bijective. (By convention, $M_0 = M$.)

Remark 3.2. *Since* Prim *is a left exact functor from P-comodules to $\mathbb{F}_p$-modules, it follows that the induced map*

$$\mathrm{Prim}(\tilde{F}_i) \to \mathrm{Prim}(\tilde{F}_{i+1})$$

is trivial.

For BP_*BP-comodules, we define minimal resolutions in terms of reductions modulo I.

Definition 7. *Let M be a BP_*BP-comodule which is free over BP_*. A cofree resolution of M is called a minimal resolution if its reduction modulo I is a minimal resolution of M/I.*

Note that a minimal resolution has the smallest size among all cofree resolutions.

4 Construction of minimal resolutions

We will construct minimal resolutions of BP_*BP-comodules by lifting minimal resolutions of reductions modulo I.

First we introduce the notion of cogenerators dual to the notion of generators. Let M be a locally finite BP_*BP-comodule which is free as BP_*-module, and let X be a locally finite free BP_*-module. A BP_*-module map

$$M \to X$$

exhibits X as cogenerators of M if the adjoint map

$$M \to BP_*BP \otimes_{BP_*} X$$

is strongly injective. In this case, any comodule map $N \to M$ is determined by its corestriction to the cogenerators $N \to M \to X$.

Proposition 8. *A map*

$$f : M \to X$$

exhibits X as cogenerators of M if and only if its reduction

$$\tilde{f} : M/I \to X/I$$

modulo I exhibits X/I as cogenerators of M/I as a P-comodule.

Proof. The adjoint map

$$M/I \to P \otimes_{\mathbb{F}_p} X/I$$

for $\tilde{f}$ is the reduction of the adjoint map for f modulo I, so the proposition follows from Proposition 3. ☐

Now we construct minimal resolutions as follows. Suppose M is a locally finite BP_*BP-comodule with free underlying BP_*-module. We will construct strongly exact sequences

$$0 \to M \to F_0 \to M_1 \to 0$$
$$0 \to M_1 \to F_1 \to M_2 \to 0$$
$$\vdots$$

such that each F_i is cofree.

Set $M_0 = M$. Suppose we have constructed a locally finite BP_*BP-comodule M_n which is free as a BP_*-module. We do the following to construct M_{n+1}:

1. Find a minimal cogenerator

$$\tilde{f}_n : M_n/I \to \tilde{X}_n$$

for M_n/I, i.e., an $\mathbb{F}_p$-module $\tilde{X}_n$ such that the adjoint map

$$M_n/I \to P \otimes \tilde{X}_n$$

is injective and the map

$$\mathrm{Prim}(M_n/I) \to \tilde{X}_n$$

is bijective.

2. Take a free BP_*-module X_n such that $X_n/I \cong \tilde{X}_n$. By the freeness of M_n as a BP_*-module, we can lift $\tilde{f}_n$ to

$$f_n : M_n \to X_n.$$

3. Take F_n to be $BP_*BP \otimes_{BP_*} X_n$, and M_{n+1} to be the cokernel of the adjoint map

$$g_n : M_n \to F_n$$

of f_n, so we have the quotient map

$$h_n : F_n \to M_{n+1}.$$

In this way, we construct a minimal resolution inductively.

Remark 4.1. *The first step is the standard one for computing Adams E_2 terms using minimal resolutions. In the second step, in addition to constructing the map f_n, we also need to do a Gaussian elimination for its adjoint map g_n, to find the quotient matrix needed in step 3. And in step 3, we need to compose the quotient map with the coaction map on F_n to get the coaction map on M_{n+1}.*

5 Optimization of the process

The complexity of computing a minimal resolution of M/I has smaller order than computing a minimal resolution of M. So the process can be optimized by computing a minimal resolution of M/I first, and then using it as a model for a resolution of M.

Once we know the structure of the minimal resolution of M/I, the structures of the cofree comodules F_i are already known. The problem with an arbitrary lift of the minimal resolution of M/I is that the compositions of consecutive maps are not guaranteed to be zero.

Once we know the structure of F_n, we already know a set of cogenerators for it. Moreover, these also make a set of cogenerators for M_n. So the data for g_n and h_n (in Step 3 of Section 4) are determined by their corestrictions to cogenerators. This reduces the order of the size of the matrices for the data of f_n and g_n. So the optimized process is as follows:

1. This is the same as Step 1 of Section 4, but it is computed beforehand.

2. This is the same as Step 2 of Section 4, but it is computed beforehand.

3. Compute the matrix for the composite map

$$F_n \to M_{n+1} \to X_{n+1}$$

using a Gaussian elimination process for the map

$$g_n : M_n \to F_n.$$

4. In order to compute g_n, we only need to know the composite

$$M_n \xrightarrow{\psi} BP_*BP \otimes M_n \to BP_*BP \otimes X_n.$$

So we only need to compute the composite

$$M_{n+1} \xrightarrow{\psi} BP_*BP \otimes M_{n+1} \to BP_*BP \otimes X_{n+1}$$

instead of the full formula for the coaction of M_n, by composing the coaction of F_n with the composite

$$F_n \to M_{n+1} \to X_{n+1}.$$

6 Computations of homology

With the minimal resolution constructed, our next step is to compute the homology of the chain complex of its primitives. We will modify the algorithm of [1].

Suppose

$$X_0 \to X_1 \to \cdots$$

is a complex of locally finite $\mathbb{Z}_p$-modules. Let each X_i be maximally filtered. We do not require that the differentials in the complex respect the filtrations. For each i, we take a set A_i consisting of an $\mathbb{F}_p$ basis for the filtration quotients of X_i. Then each A_i has a canonical order.

Remark 6.1. *Here we allow the generality that the filtration is indexed by any ordinal number. In actual computations we will use appropriate truncations to make the filtration finite.*

A Curtis table is a list with entries of the form:

- a

  where $a \in A_i$ for some i, or

- $a \to b$

  where $a \in A_i$ and $b \in A_{i+1}$ for some i;

such that

1. an entry a is in the table if and only if a is the leading term of a cycle, and no boundary has leading term a.

2. an entry $a \to b$ is in the table if and only if

   (a) there is an element $x \in X_i$ with leading term a, and $d(x)$ has leading term b, and

   (b) for any element x' such that $d(x')$ has leading term b, the leading term of x' is at least a.

The following is proved in [4].

Proposition 9. *Suppose*

$$X_0 \to X_1 \to \cdots$$

is a complex of locally finite $\mathbb{Z}_p$-modules such that each X_i is maximally filtered, and let A_i be an $\mathbb{F}_p$ basis for the filtration quotients of X_i. Then a Curtis table exists and is unique.

Now we suppose there is an additional filtration $\mathcal{F}_i$ on each X_i which is preserved by the differentials. Moreover, we suppose that the maximal filtration associated to A_i is a refinement of $\mathcal{F}_i$. Then the Curtis table describes the structure of the spectral sequence defined by $\mathcal{F}_i$ (see [4] for details).

Proposition 10. *The entries of the form*

$$a$$

in the Curtis table correspond bijectively to the surviving permanent cycles in the spectral sequence defined by the filtration $\mathcal{F}_i$.

The entries of the form

$$a \to b$$

correspond bijectively to the differentials in the spectral sequence, where the length of the differential is the difference of the $\mathcal{F}$-filtration degrees of a and b. (We include all the differentials from d_0.)

7 The algebraic Adams-Novikov spectral sequence

Let

$$M \to F_0 \to F_1 \to \cdots$$

be a cofree resolution of a BP_*BP-comodule M.

Then $\mathrm{Ext}(M)$ is computed by the complex

$$\mathrm{Prim}(F_0) \to \mathrm{Prim}(F_1) \to \cdots .$$

By construction, when M is free as a BP_*-module, $\mathrm{Prim}(F_i)$ has the structure of a free BP_*-module when a set of cogenerators for F_i is given. (Note that this BP_*-module structure is not preserved by the differentials.)

The algebraic Adams-Novikov filtration is given by filtering $\mathrm{Prim}(F_i)$ by powers of the augmentation ideal I. To be precise, let G_i be a set of BP_*-generators for $\mathrm{Prim}(F_i)$. Then all elements of $\mathrm{Prim}(F_i)$ are linear combinations of expressions of the form $v_0^{i_0} v_1^{i_1} \cdots v_k^{i_k} a$ with $a \in G_i$. By convention, $v_0 = p$. Then we define the decreasing algebraic Adams-Novikov filtration by setting $v_0^{i_0} v_1^{i_1} \ldots v_k^{i_k} a$ to have filtration $i_0 + i_1 + \cdots + i_k$.

Let A_i be the set

$$\{v_0^{i_0} v_1^{i_1} \ldots v_k^{i_k} a : i_j \geq 0, a \in G_i\}$$

We order A_i by the following rules. First order them using the Adams-Novikov filtration. Then, amongst elements with the same Adams-Novikov filtration, use a lexicographic order.

By Proposition 10, we have

Proposition 11. *The Curtis table associated to the above A_i gives the structure of the algebraic Adams-Novikov spectral sequence for M.*

Remark 7.1. *We can also introduce an order, by taking the lexicographic order directly. This results in the Bockstein spectral sequence.*

8 The Atiyah-Hirzebruch spectral sequence and multiplicative structure

Now we fix a minimal resolution

$$BP_* \to F_0 \to F_1 \to \cdots$$

for BP_*.

Let M be a BP_*BP comodule with free underlying BP_*-module. Recall that the tensor product of two BP_*BP-comodules has the structure of a BP_*BP-comodule (see [3] A1.1.2). We have

$$M \otimes BP_* \cong M.$$

Since F_i is cofree, $M \otimes F_i$ is also cofree, and

$$\mathrm{Prim}(M \otimes F_i) \cong M \otimes_{BP_*} \mathrm{Prim}(F_i).$$

Now suppose M is filtered as a BP_*BP-comodule. Then there is an associated algebraic Atiyah-Hirzebruch spectral sequence computing $\mathrm{Ext}(M)$. We have the following standard fact for Atiyah-Hirzebruch differentials.

Proposition 12. *Let M have a two-step filtration as shown in the short exact sequence*

$$0 \to BP_* \to M \to \Sigma^j BP_* \to 0.$$

Let $h \in \mathrm{Ext}^{j,1}(BP_*)$ be the element corresponding to this extension. Then the Atiyah-Hirzebruch differentials for M corresponds to multiplication by h.

So we can compute multiplication by elements in homological degree 1 by computing the Atiyah-Hirzebruch differentials. To do this, we do the following.

We select a set K of BP_*-generators for M, with order refining the filtration on M. Then an element in

$$\mathrm{Prim}(M \otimes F_i) \cong M \otimes_{BP_*} \mathrm{Prim}(F_i)$$

is a linear combination of expressions of the form

$$k \otimes v_0^{i_0} \ldots v_k^{i_k} a$$

with $k \in K$ and $a \in G_i$. Here G_i is a set of BP_*-generators for $\mathrm{Prim}(F_i)$, as in Section 7.

We order these elements as follows. First consider the filtration of k, then the Adams-Novikov filtration of $v_0^{i_0} \dots v_k^{i_k}$, and finally the lexicographic order. By Proposition 10, the Curtis table for this order gives the structure of the Atiyah-Hirzebruch spectral sequence for M.

9 Source Code

The source code is written in C++ (using C++11 standard) and uses the libraries OpenMP (Open Multi-Processing) and GMP (GNU Multiple Precision Arithmetic Library). It has been tested on linux platforms, compiled with GCC (GNU Compiler Collection).

9.1 algebra.h

The file `algebra.h` is a set of templates for basic algebraic structures.

Abelian groups: Abelian groups are required to have addition, negation, as well as IO operations.

Rings: Rings are abelian groups with multiplication. We also require an operation to produce inverses of invertible elements.

9.2 modules.h

The file `modules.h` consists of operations on modules over a ring.

Vectors: The class `vectors` is the data structure to store vectors with entries in a ring.

Module operations: The class `ModuleOp` gives operations on R-modules.

Addition: There are two types of addition: one is ordinary, and one uses the move operation to optimize for the cases when the arguments are no longer needed.

IO operations

Termwise operations

Searching operations

Sums: The function `sum` computes the sum of an array of elements using a parallel algorithm.

Swapping operation: The function `swapping` swaps the inside and outside indices of a vector of vectors.

Free modules: The class `modules` gives the data for a graded free module.

9.3 matrices.h

The file `matrices.h` contains functions on matrices.

Matrices: The class `matrix` is a template for matrices.

Initializing matrices

IO operations

Compositions

Quotients

Modifying matrices

Sorting entries

Curtis tables

9.4 hopf_algebroid.h

The file `hopf_algebroid.h` contains functions related to Hopf algebroids and comodules.

Comodules over a Hopf algebroid

Cofree comodules: Recall that a comodule is cofree over (R, Γ) if it is a direct sum of copies of shifts of Γ.

Operations for Hopf algebroids: The class `Hopf_Algebroid` contains operations related to a Hopf algebroid.

Resolution of comodules: In getting a cofree resotution, we first construct a series of short exact sequences, and then combine the short exact sequences into a long exact sequence.

Resolution without models

Resolution of comodules using a model

Combining the short exact sequences

Operations on comodules

Adjoint maps to the projections

Embedding comodules into cofree ones

Quotient comodules

Computations of multiplicative structure on resolutions: We secretly make a cofree comodule with specified generators into a module over the algebroid, and compute the scalar product with the algebroid.

Initializations

Operations on cofree comodules

9.5 matrices_mem.h

The file `matrices_mem.h` realizes the abstract class `matrix`, such that the data of the matrix is stored in memory.

Matrix class

Gaussian elimination

Curtis tables

9.6 polynomial.h

The file `polynomial.h` contains templates for operations on polynomials.

The polynomial class: The class `PolynomialOp` collects operations for polynomials.

Multiplications

Multiplication with parallel algorithms: The class `PolynomialOp_Para` uses parallel algorithms for multiplications.

Other operations

9.7 exponents.h and exponents.cpp

The files `exponents.h` and `exponents.cpp` use an additive hash table to deal with multi-variable polynomials.

9.8 Fp.h and Fp.cpp

The files `Fp.h` and `Fp.cpp` realize the operations on the ring $\mathbb{F}_p$.

Operations: The class `Fp_Op` collects operations on $\mathbb{F}_p$.

9.9 mon_index.h and mon_index.cpp

The files `mon_index.h` and `mon_index.cpp` contain the data for the set of monomials appearing in the dual Steenrod algebra.

9.10 steenrod.h and steenrod.cpp

We realize the Hopf algebra of the dual Steenrod algebra to generate the minimal resolutions modulo v_i's. The files `steenrod.h` and `steenrod.cpp` contain operations for the Hopf algebra of the dual Steenrod algebra.

9.11 steenrod_init.h and steenrod_init.cpp

The files `steenrod_init.h` and `steenrod_init.cpp` contain the initializations for the Steenrod algebra.

9.12 Qp.h and Qp.cpp

We construct the ring $\mathbb{Z}[\frac{1}{p}]$, which serves as the coefficient ring for the Hazewinkel generators.

9.13 BPQ.h and BPQ.cpp

We first construct the Hopf algebroid BP_*BP over the rationals, using the Hazewinkel generators. These generators have the advantage that all denominators are powers of p. We use the logarithm to get the formula for the right action and left action for the Hopf algebroid BP_*BP over the rationals, and then use the Hazewinkel generators to transform to the v_i generators.

9.14 Z2.h and Z2.cpp

We use $\mathbb{Z}/2^{64}$ to play the role of the 2-adic numbers. This suffices as long as we do not meet with elements of order greater than 2^{64}. As the E_2 term of the Adams-Novikov spectral sequence is torsion in positive degrees, we are alerted by the vanishing of a 2-Bockstein that there is a flow-out issue. We use that unsigned integers are automatically truncated when flowing-out in the C++ standard.

References

[1] Edward B. Curtis, Paul Goerss, Mark Mahowald, and James R. Milgram, *Calculations of unstable Adams E_2 terms for spheres*, in Algebraic topology (Seattle, Wash., 1985), 208–266, Lecture Notes in Math. **1286**, Springer, 1987.

[2] Bogdan Gheorghe, Guozhen Wang and Zhouli Xu, *The special fiber of the motivic deformation of the stable homotopy category is algebraic*, preprint, arXiv:1809.09290 (2018).

[3] Douglas C. Ravenel, *Complex cobordism and stable homotopy groups of spheres*, Pure and Applied Mathematics **121**, Academic Press, 1986.

[4] Guozhen Wang and Zhouli Xu, *The algebraic Atiyah-Hirzebruch spectral sequence of real projective spectra*, preprint, arXiv:1601.02185 (2016).