ADAMS DIFFERENTIALS VIA THE SECONDARY STEENROD ALGEBRA

DEXTER CHUA

Abstract. Using the secondary Steenrod algebra, we compute all Adams $d_2$ differentials of the sphere up to the 140\textsuperscript{th} stem, and all $d_2$'s of $C_2$, $C\eta$, $C\nu$, $C\sigma$ and $C_2 \wedge C\sigma$ up to the 112\textsuperscript{th} stem.

This data allows us to compute hidden $2, \eta, \nu$ and $\sigma$ extensions of the sphere that jump by one filtration, from which we resolve all remaining unknown $d_2$, $d_3$, $d_4$ and $d_5$ differentials of the sphere up to the 95\textsuperscript{th} stem.

Contents

Part 1. Computer calculation of $d_2$ differentials 4

Conventions 4

1. Secondary homological algebra 4

2. Computation of the secondary differential 10

3. The secondary Steenrod algebra 13

4. Implementation details 15

Part 2. Computing hidden extensions and differentials 18

Conventions 18

5. Differentials and hidden extensions 19

6. Computation of new differentials 22

7. Tables 23

Appendix A. Source code and raw data 25

References 27

We work exclusively at the prime 2.

In [BJ11], Baues and Jibladze introduced an algorithm to compute $d_2$ differentials in the Adams spectral sequence of the sphere based on the secondary Steenrod algebra. They implemented the algorithm and used it to compute all $d_2$'s up to $t = 35$, all of which were already known at the time.

In this paper, we extend their work in two directions. First, we compute all $d_2$'s up to the 140\textsuperscript{th} stem, many of which were previously unknown. Second, we generalize the algorithm to compute $d_2$'s for any spectrum. We use this to compute all $d_2$'s for the following spectra up to the 112\textsuperscript{th} stem:

$C_2$, $C\eta$, $C\nu$, $C\sigma$, $C_2 \wedge C\eta$.

As usual, $C\alpha$ denotes the cofiber of a map $\alpha: \mathbb{S}^n \to \mathbb{S}$. 

The differentials in $C\alpha$ provide valuable information, even to those who only care about the sphere — they correspond to hidden $\alpha$-extensions in the sphere. In turn, a generalized Leibniz rule lets us use hidden extensions to relate differentials of different lengths. Using this, we compute a number of longer differentials in the sphere. In particular, we resolve all unknown classical Adams $d_3$, $d_4$ $d_5$ differentials up to the 95th in [IWX20b]. We also correct two mistakes in [IWX20b] (Remark 6.3).

**Comparison with prior work.** Since the source code of Baues and Jibladze’s implementation is not available, we cannot tell for sure why we are able to compute $d_2$’s further than them. However, we believe a major factor is that we used a simpler presentation of the secondary Steenrod algebra.

Unlike the ordinary one, the secondary Steenrod algebra is a differential graded algebra. Thus, it admits multiple different presentations. The original presentation by [Bau06] was a sort of free presentation, which is guaranteed to exist but is somewhat unwieldy. We use a simpler presentation by Nassau from [Nas12], which is both smaller and easier to implement, and we expect this to translate to a substantial performance improvement.

In [BJ11], they restricted to the case of the sphere presumably because there was no obvious way to compute the secondary cohomology of a general spectrum. We sidestep this issue by classifying all possible lifts of a Steenrod module to a secondary Steenrod module. It turns out to be a torsor over $\text{Ext}^{2,1}(H^*X, H^*X)$. In all cases of interest, there is a unique lift. Even if it is not unique, we can simply pick an arbitrary lift. If two lifts differ by $\chi \in \text{Ext}^{2,1}(H^*X, H^*X)$, then the $d_2$ differentials computed differ by multiplication-by-$\chi$. If $\text{Ext}^{2,1}(H^*X, H^*X)$ is finite, which is the case when $H^*X$ itself is, then we can determine all $d_2$’s once we know a finite number of them.

The final section uses hidden extensions and a generalized Leibniz rule to propagate differentials. The gists of these arguments are well-understood, and ad hoc versions have been employed in e.g. [IWX20b]. The main novelty is in using synthetic spectra to define hidden extensions of non-permanent classes, which lets us prove general statements relating hidden extensions and differentials.

**Useful references.** Most prior work was done by Baues and his collaborators. We found [BJ06; BJ11, Chapters 1–3] to be the most useful references. The book [Bau06] is the only source that describes the original presentation of the secondary Steenrod algebra, but we do not make use of this information; we use the presentation of [Nas12] instead (which in turn uses [Bau06] itself).

**Future directions.** We assume the reader is familiar with synthetic spectra; see the beginning of Section 5 for a recap.

This paper is a compromise — we milk as much out of the algorithm as possible without having to develop significant new theory. We hope to develop the theory “properly” in a future paper.

The correct theorem to prove is an isomorphism of categories between $\text{Mod}_{C^T}$ and modules over the secondary Steenrod algebra. This is not literally true, since $\text{Mod}_{C^T}$ is based on the dual Steenrod algebra. Nevertheless, this statement should

---

1Some of these differentials have been independently computed in unpublished work of Burklund–Isaksøen–Xu. Specifically, they have computed the differentials on $\Delta^2g_2$, $h_1\Delta^2g_2$, $h_0^3\Delta h_2$, $x_{95,7}$, $\Delta^2Mh_1$, and $\Delta^2MH_1$. 
be true when restricted to finite spectra, and is true in general once we put \(\text{Ind} \) in the right places. Such a theorem would be immensely useful. Firstly, it would let us extend our algorithms to modified Adams spectral sequences and apply it to, say, \(C(2, v_1^1)\). More importantly, it would let us completely calculate \(\pi_\ast C\tau^2\) as a ring. The most obvious benefit is that we can now compute hidden extensions by any element on the \(E_3\) page, not just filtration one elements. For example, products by \(\kappa\) and \(\bar{\kappa}\) are fairly useful.

A more subtle improvement is that it replaces the indeterminacy in hidden extensions with having to compute the “total differential” \(\delta x \mod \tau\) instead of \(\tau\). While we have absolutely no control over the former, we can sometimes get our hands on the latter.

To illustrate this, we compute \(d_3(h^0_3 h^5_5)\) from \(d_2(h^5_5) = h^0_0 h^2_4\). Let \(\theta_4\) be any lift of \(h^2_4\) to \(C\tau^2\). Then the differential \(d_2(h^5_5) = h^0_0 h^2_4\) tells us
\[
\delta h^5_5 = 2\theta_4 \mod \tau.
\]
The only element in bidegree \((30, 3)\) is \(h^2_0 \theta_4 = 2^2 \theta_4\). So we can write
\[
\delta h^5_5 = 2\theta_4 + a_0 \tau \theta_4 \mod \tau^2,
\]
for some coefficient \(a_0\). So
\[
(\delta(h^3_0 h^5_5) = 2^3 \delta h^5_5 = 2^2 \theta_4 + a_0 \tau \theta_4 \mod \tau^2.
\]
Machine computation would tell us \(2^4 \theta_4 = \tau h^0_0 \Delta h^2_2\), so \(d_3(h^3_0 h^5_5) = h^0_0 \Delta h^2_2\).

If we attempted to apply the results of Section 5 directly, we only know that \(d_2(h^3_0 h^5_5)\) is a hidden 2-extension of \(d_2(h^3_0 h^5_5) = h^3_0 h^5_5\). But \(h^0_0 \Delta h^2_2\) is in the indeterminacy, so this is not helpful.

The point is that \(d_2(h^3_0 h^5_5)\) only captures the value of \(\delta(h^3_0 h^5_5)\) mod \(\tau\), but we know more about \(\delta(h^3_0 h^5_5)\) — it is 2-divisible. We can use this extra knowledge to pin down the exact value of \(\delta(h^3_0 h^5_5)\) to finish the computation.

The same technique can be used to obtain all the image of \(J\) differentials.

**Overview.** The paper is divided into two parts, which are fairly different in flavour and can be read independently.

**Part 1.** We cover the background needed to compute \(d_2\) differentials. The high-level goal is to provide the reader with the ability to independently implement the algorithm without referencing external sources if they so desire. Note that we do not perform the hardest part of the whole story, which is to prove that this algorithm we write down actually computes Adams \(d_2\)'s!

In Section 1, we provide an exposition of secondary homological algebra using the language of quasi-categories. From this, we classify possible lifts of Steenrod modules to secondary ones. This forms the theoretical basis of the Baues–Jibladze algorithm.

In Section 2, we translate the higher categorical language into a concrete algorithm for computing \(d_2\)'s. This version is less amenable to high-level reasoning but translates directly to code.

In Section 3 we fill in the final missing bit by describing Nassau’s presentation \([Nas12]\) of the secondary Steenrod algebra.
In Section 4 we describe the real-world aspects of the algorithm and design decisions in the implementation. We end the section by analyzing the performance of our implementation.

**Part 2.** We use the computer-calculated $d_2$ differentials to compute longer differentials in the sphere.

Section 5 explains the relations between hidden extensions and differentials. This comprises of two main lemmas — differentials in $C\alpha$ gives hidden $\alpha$-extensions, and a Leibniz rule for hidden extensions.

Section 6 uses the results of the previous section to resolve unknown differentials in [IWX20b]. This section only includes the applications that do not follow immediately from Leibniz’s rule; the rest are listed in the tables in Section 7.

Appendix A discusses usage of the program and raw data available.

**Acknowledgements.** All of this would not have been possible without Hood Chatham, who introduced me to the world of computer Ext calculations and pioneered the ext-rs library that the implementation is based on. The material in Part 2 was born out of conversations with Robert Burklund, from whom I learnt the art of synthetic spectra.

I would also like to thank Christian Nassau, Dan Isaksen, John Rognes, Mike Hopkins and Robert Bruner for their encouragement and insightful conversations throughout the project. Martin Frankland also provided helpful comments on an earlier draft.

**Part 1. Computer calculation of $d_2$ differentials**

**Conventions**

- $\text{Ch}(\text{Ab}^\ast)$ is the $\infty$-category of graded chain complexes over $\mathbb{Z}$. We use $\Sigma$ to denote the categorical suspension (shifting in the chain complex direction) and $[1]$ to denote the internal degree shift within $\text{Ab}^\ast$.
- If $X \in \text{Ch}(\text{Ab}^\ast)$, then $\pi_t X$ is the $t^{th}$ homotopy group of $X$, which is a graded abelian group. Some may prefer to call this the homology group.
- We make heavy use of Massey products in Section 1. It is often the case that we have chosen null-homotopies of certain compositions, and our Massey product will be computed using these specific null-homotopies. Thus, the Massey product will be an actual element instead of a subset.

**1. Secondary homological algebra**

Computing the Adams $E_2$ page requires doing homological algebra over the Steenrod algebra, which largely involves taking free resolutions in the 1-category of Steenrod modules.

To compute the Adams $E_3$ page, we need to do “secondary homological algebra” over the secondary Steenrod algebra, which is the focus of this section. This involves taking free resolutions in the 2-category of pair modules over the secondary Steenrod algebra.

**Definition 1.1** ([BJ11, Definition 1.1.5]). A pair algebra is an $E_1$-ring $B$ in $\text{Ch}(\text{Ab}^\ast)$ with an element $\tau \in \pi_1 B$ of degree 1 such that

$$\pi_1 B \cong \pi_0 B[\tau]/\tau^2.$$
In particular, $\mathcal{B}$ is 1-truncated.

The category $\text{Mod}_{\Sigma}^B$ of pair modules over $\mathcal{B}$ is the subcategory of $\text{Mod}_B$ consisting of left $E_1$-modules $M$ such that

$$\pi_* M = \pi_* \mathcal{B} \otimes_{\pi_0 B} \pi_0 M = \pi_0 M[\tau]/\tau^2.$$ 

**Remark 1.2.** These definitions are stronger than those in [BJ11], and are heavily inspired by [PV19]. In the language of [PV19], a pair algebra behaves like a 1-truncation of a shift algebra, and a pair module is a potential 1-stage.

In this section, we fix a pair algebra $\mathcal{B}$. We observe that

**Lemma 1.3.** $\text{Mod}_{\Sigma}^B$ is a 2-category.

**Lemma 1.4.** Let $M, N \in \text{Mod}_{\Sigma}^B$. Suppose $M$ is free, i.e. it is a direct sum of modules of the form $B[k]$. Then

$$[M, N] = \text{Hom}_{\pi_0 B}(\pi_0 M, \pi_0 N).$$

**1.1. Secondary chain complexes.** The notion of a chain complex in a higher category requires more care.

**Definition 1.5** ([BJ06, Definitions 2.6, 2.8]). Let $C$ be a pointed 2-category. A (secondary) chain complex in $C$ is a sequence

$$R_0 \xleftarrow{d_4} R_1 \xleftarrow{d_2} R_2 \xleftarrow{d_3} R_3 \xleftarrow{d_4} R_4 \xleftarrow{d_5} \cdots$$

together with specified null-homotopies of $d_{k-1}d_k$, such that all three-fold Massey products $\langle d_{k-2}, d_{k-1}, d_k \rangle$ vanish.

The category of chain complexes $\text{Ch}(C)$ is a full subcategory of the category of commutative diagrams of the form

$$\begin{array}{cccccccc}
R_0 & \xleftarrow{d_1} & R_1 & \xleftarrow{d_2} & R_2 & \xleftarrow{d_3} & R_3 & \xleftarrow{d_4} & \cdots \\
\downarrow f_0 & & \downarrow f_1 & & \downarrow f_2 & & \downarrow f_3 & & \downarrow f_4 \\
S_0 & \xleftarrow{d_1'} & S_1 & \xleftarrow{d_2'} & S_2 & \xleftarrow{d_3'} & S_3 & \xleftarrow{d_4'} & \cdots 
\end{array}$$

**Remark 1.6.** In $\text{Mod}_{\Sigma}^B$, the Massey product of $A \to B \to C \to D$ is an element in $[\Sigma A, D] = \text{Hom}_{\pi_0 B}(\pi_0 A, \pi_1 D)$.

**Remark 1.7.** This definition of a chain complex is only correct in a 2-category. In general, we need to require higher-order compositions to vanish as well.

If $C$ is in fact a 1-category, then a secondary chain complex is the usual notion of a chain complex, i.e. a sequence of maps where consecutive maps compose to zero.

**Definition 1.8.** The category of homotopy chain complexes in $C$ is $\text{Ch}(hC)$.

For us, a (homotopy) chain complex always refers to one in $\text{Mod}_{\Sigma}^B$.

**Remark 1.9.** Concretely, a morphism $R_* \to S_*$ of homotopy chain complexes is a collection of maps $f_q: R_q \to S_q$ such that the diagram

$$\begin{array}{cccccccc}
R_0 & \xleftarrow{d_1 R} & R_1 & \xleftarrow{d_2 R} & R_2 & \xleftarrow{d_3 R} & R_3 & \xleftarrow{d_4 R} & \cdots \\
\downarrow f_0 & & \downarrow f_1 & & \downarrow f_2 & & \downarrow f_3 & & \downarrow f_4 \\
S_0 & \xleftarrow{d_1 S} & S_1 & \xleftarrow{d_2 S} & S_2 & \xleftarrow{d_3 S} & S_3 & \xleftarrow{d_4 S} & \cdots 
\end{array}$$
is homotopy commutative.

A morphism of chain complexes is a morphism of homotopy chain complexes with chosen homotopies of the squares such that the matrix Massey products

\[
\left( \begin{array}{ccc}
-d_{s-2}^R & 0 & 0 \\
-d_{s-2}^R & 0 & 0 \\
-f_{s-1}^S & d_{s-1}^S & d_{s+1}^S
\end{array} \right)
\]

all vanish. Alternatively, this says the total chain complex of this bicomplex in \( \text{Ch}(\text{Ab}_s) \) is also a chain complex.

Note that the Massey product is of the form

\[
\langle (d_{s-2}^R, d_{s-1}^R, d_s^R), (d_{s-1}^S, d_s^S, d_{s+1}^S) \rangle
\]

the diagonal entries are necessarily zero by definition of a chain complex. The requirement is that \( \gamma_s \) also vanishes.

Informally, this says that the following two null-homotopies of \( f_{s-2}d_{s-1}^Rd_s^R \) are equal:

1. Take the null-homotopy of \( d_{s-1}^Rd_s^R \) and compose it with \( f_{s-2} \).
2. Use the given homotopy from \( f_{s-2}d_{s-1}^Rd_s^R \) to \( d_{s-1}^Sd_s^Sf_s \), then apply the null-homotopy of \( d_{s-1}^Sd_s^S \).

1.2. Secondary resolutions.

Definition 1.10. A (homotopy) chain complex \( R_* \) is exact at \( s \) if \( \pi_0R_* \) (and hence \( \pi_1R_* \)) is exact at \( s \). It is exact if it is exact at all \( s \).

Definition 1.11. Let \( M \in \text{Mod}^\otimes_B \). A secondary resolution of \( M \) is an exact chain complex \( R_* \) such that \( R_0 = M \) and \( R_s \) is free for \( s > 0 \).

We usually shift the (homological) grading so that \( R_{-1} = M \).

To obtain secondary resolutions, start with a free resolution of \( \pi_0M \) in \( \text{Mod}^\otimes_{\pi_0B} \), and lift it to a secondary chain complex. By Lemma 1.4, we can always lift it to a homotopy chain complex.

Theorem 1.12. Let \( R_* \) be a homotopy chain complex in \( \text{Mod}^\otimes_B \). Suppose

1. \( R_* \) is exact at \( \bullet \geq 1 \);
2. \( \pi_0R_* \) is projective over \( \pi_0B \) for \( s > 3 \); and
3. \( 0 \in \langle d_1, d_2, d_3 \rangle \).

Then \( R_* \) lifts to a chain complex, i.e. there is a choice of null-homotopies of \( d_{s-1}d_s \) such that \( \langle d_{s-2}, d_{s-1}, d_s \rangle = 0 \) for all \( s \).

Proof. Pick the null-homotopies of \( d_1d_2 \) and \( d_2d_3 \) such that \( \langle d_1, d_2, d_3 \rangle = 0 \).

Inductively, suppose we have chosen the null-homotopies of \( d_{k-1}d_k \) such that \( \langle d_{k-2}, d_{k-1}, d_k \rangle = 0 \) for \( k \leq s \).

Pick any null-homotopy of \( d_sd_{s+1} \). Then

\[
d_{s-2}\langle d_{s-1}, d_s, d_{s+1} \rangle = \langle d_{s-2}, d_{s-1}, d_s \rangle d_{s+1} = 0.
\]

Since \( \pi_1R_* \) is exact and \( \pi_0R_{s+1} \) is projective, we know that \( \langle d_{s-1}, d_s, d_{s+1} \rangle \) factors through \( \pi_1d_{s-1} \). Adding the factorization to the null-homotopy of \( d_sd_{s+1} \) kills the Massey product.

Corollary 1.13. If \( R_* \) is an exact homotopy chain complex such \( \pi_0R_* \) is projective over \( \pi_0B \) for \( s > 2 \), then \( R_* \) lifts to a chain complex.
Proof. Shift the chain complex to the right by 1 and set \( R_0 = 0 \). Then the last condition of Theorem 1.12 is automatic.

Corollary 1.14 ([BJ06, Lemma 2.14]). Any \( M \in \text{Mod}_{R}^\Sigma \) admits a secondary resolution. In fact, any free resolution of \( \pi_0 M \) lifts to a secondary resolution of \( M \).

Proof. Let \( R_* \) be a free resolution of \( \pi_0 M \) in \( \text{Mod}_{\pi_0 R}^\Sigma \). By Lemma 1.4, this lifts to a homotopy chain complex in \( \text{Mod}_{R}^\Sigma \). By Corollary 1.13, this further lifts to a chain complex.

Theorem 1.15 ([BJ06, Lemma 2.15]). Let \( R_* \) and \( S_* \) be exact chain complexes in \( \text{Mod}_{R}^\Sigma \), and suppose \( R_s \) is free for \( s \geq 2 \). Let \( f_* : R_* \rightarrow S_* \) be a map of homotopy chain complexes. Then the \( f_* \) lift to a map of chain complexes.

Proof. We have the following diagram that commutes up to homotopy.

\[
\begin{array}{cccc}
R_{s-3} & \overset{d_{s-2}}{\leftarrow} & R_{s-2} & \overset{d_{s-1}}{\leftarrow} & R_{s-1} & \overset{d_s}{\leftarrow} & R_s \\
\downarrow f_{s-3} & & \downarrow f_{s-2} & & \downarrow f_{s-1} & & \downarrow f_s \\
S_{s-3} & \overset{d_{s-2}}{\leftarrow} & S_{s-2} & \overset{d_{s-1}}{\leftarrow} & S_{s-1} & \overset{d_s}{\leftarrow} & S_s
\end{array}
\]

For this to become a map of chain complexes, we need to pick the homotopies of each square such that for every \( s \), the matric Massey product

\[
\begin{pmatrix}
0 & 0 \\
\gamma & 0
\end{pmatrix} = \begin{pmatrix}
-d_{s-2}^{R} & 0 \\
-f_{s-2} & d_{s-1}^{S}\end{pmatrix}, \begin{pmatrix}
-d_{s-1}^{R} & 0 \\
-f_{s-1} & d_s^{S}
\end{pmatrix}, \begin{pmatrix}
-d_s^{R} & 0 \\
-f_s & d_{s+1}^{S}
\end{pmatrix}
\]

vanishes.

We assume we have chosen homotopies of the maps \( R_k \rightarrow S_{k-1} \) up till \( k = s-1 \), and the above Massey product vanishes for such \( k \). Again pick a homotopy for \( R_s \rightarrow R_{s-1} \) arbitrarily. By induction, we know that

\[
\begin{pmatrix}
-d_{s-3}^{R} & 0 \\
-f_{s-3} & d_{s-2}^{S}
\end{pmatrix}, \begin{pmatrix}
-d_{s-2}^{R} & 0 \\
-f_{s-2} & d_{s-1}^{S}
\end{pmatrix}, \begin{pmatrix}
-d_{s-1}^{R} & 0 \\
-f_{s-1} & d_s^{S}
\end{pmatrix}, \begin{pmatrix}
-d_s^{R} & 0 \\
-f_s & d_{s+1}^{S}
\end{pmatrix}
\]

\[
= \begin{pmatrix}
-d_{s-3}^{R} & 0 \\
-f_{s-3} & d_{s-2}^{S}
\end{pmatrix}, \begin{pmatrix}
-d_{s-2}^{R} & 0 \\
-f_{s-2} & d_{s-1}^{S}
\end{pmatrix}, \begin{pmatrix}
-d_{s-1}^{R} & 0 \\
-f_{s-1} & d_s^{S}
\end{pmatrix}, \begin{pmatrix}
-d_s^{R} & 0 \\
-f_s & d_{s+1}^{S}
\end{pmatrix}
\]

So \( d_{s-2}^S \gamma = 0 \), and \( \gamma \) lifts through \( d_{s-1}^S \) to a map \( \pi_0 R_s \rightarrow \pi_1 S_{s-1} \). Adding this to the homotopy of the \( R_s \rightarrow S_{s-1} \) square kills the matric Massey product.

Corollary 1.16. Given any two secondary resolutions \( R_*, S_* \) of \( M \), there is a morphisms between them covering the identity on \( M \).

The morphism should be thought of as a quasi-isomorphism, but we do not have the language to state that.

Proof. The standard 1-categorical argument gives us a homotopy chain map, and Theorem 1.15 lifts it to a genuine chain map.

This allows us to define the secondary differential.

Definition 1.17. Let \( M, N \in \text{Mod}_{R}^\Sigma \). Define the secondary differential

\[
d_2 : \text{Ext}_{\pi_0 R}^{s-t}(\pi_0 M, \pi_0 N) \rightarrow \text{Ext}_{\pi_0 B}^{s+2,t+1}(\pi_0 M, \pi_0 N)
\]

as follows:
Pick a secondary resolution $R_\bullet$ of $M$, shifted so that $M = R_{-1}$. A class in $\text{Ext}^{s,t}_{\pi_0 \mathcal{B}}(\pi_0 M, \pi_0 N)$ is given by a map $\pi_0 R_s \to \pi_0 N[-t]$, which lifts to a map $c: R_s \to N[-t]$. Then $d_2$ of this class is defined to be

$$(c, d_{s+1}, d_{s+2}) \in \text{Hom}_{\pi_0 \mathcal{B}}(\pi_0 R_s, \pi_0 N[-t]) = \text{Hom}_{\pi_0 \mathcal{B}}(\pi_0 R_s[1], \pi_0 N[-t]).$$

This is in the Ext group because $(c, d_{s+1}, d_{s+2})d_{s+3} = c(d_{s+1}, d_{s+2}, d_{s+3}) = 0$. There is no multiple-of-$c$ indeterminacy because we have fixed the null-homotopy of $d_{s+1}d_{s+2}$, while the multiple-of-$d_{s+2}$ indeterminacy gives us the quotient in the definition of Ext.

The main theorem that connects this to the Adams spectral sequence is

**Theorem 1.18** ([BJ11, Theorem 3.1.1]). Let $\mathcal{B}$ be the secondary Steenrod algebra, as defined in Section 3. Then there is a functor $\mathcal{H}: \text{Sp} \to \text{Mod}_\Sigma^\mathcal{B}$ such that

- there is a natural isomorphism $\pi_0 \mathcal{H}^* X = H^* X$; and
- the secondary differential on $\text{Ext}(\mathcal{H}^* X, \mathcal{H}^* Y)$ coincides with the Adams $d_2$ for $[Y, X]$.

### 1.3. Secondary refinements

Suppose we have a spectrum $X$, and we know $H^* X$ as a Steenrod module. Then $\mathcal{H}^* X$ is some pair module such that $\pi_0 \mathcal{H}^* X = H^* X$. The goal of this section is to understand the set of such pair modules.

**Definition 1.19.** Let $\bar{M}$ be a $\pi_0 \mathcal{B}$-module. A secondary refinement of $\bar{M}$ is a pair module $M$ with an isomorphism $\pi_0 M \cong \bar{M}$.

The strategy to producing secondary refinements is as follows — pick a free resolution of $\bar{M}$, lift the resolution to a chain complex in $\text{Mod}_\Sigma^\mathcal{B}$, and see what this is a resolution of. To do so, we need to be able to recover a pair module from its free resolution.

**Lemma 1.20.** If $R_\bullet$ is a free resolution of $M \in \text{Mod}_\Sigma^\mathcal{B}$, then $M$ can be recovered from the diagram

$$
\begin{array}{ccc}
R_2 & \xrightarrow{d_2} & R_1 \\
\downarrow & & \downarrow d_1 \\
0 & \xrightarrow{d_1} & R_0
\end{array}

(\dagger)

$$

i.e. from the maps $d_1, d_2$ and the null-homotopy of the composite $d_1d_2$.

More precisely, there is a functor from the category of such diagrams to $\text{Mod}_\mathcal{B}$ such that the image of a free resolution of $M$ is naturally isomorphic to $M$.

Compare this with the case of ordinary free resolutions, where we only need the first differential $d_1$.

**Proof.** Define the functor as follows — let $\Sigma^{-1} C_1$ be the fiber of $d_1$. Then the diagram gives a lift $R_2 \to \Sigma^{-1} C_1$. Let $\Sigma^{-2} C_2$ be the fiber of $R_2 \to \Sigma^{-1} C_1$. The functor then sends the diagram to $\tau_{\leq 1} C_2$.

$$
\begin{array}{ccc}
\Sigma^{-2} C_2 & \xrightarrow{i_2} & R_2 \\
\downarrow d_2 & & \downarrow d_2 \\
\Sigma^{-1} C_1 & \xrightarrow{i_1} & R_1 \\
\downarrow d_1 & & \downarrow d_1 \\
C_0 & \xrightarrow{i_0} & R_0
\end{array}

(\dagger)

$$

Define the diagram as follows — let $\Sigma^{-1} C_1$ be the fiber of $d_1$. Then the diagram gives a lift $R_2 \to \Sigma^{-1} C_1$. Let $\Sigma^{-2} C_2$ be the fiber of $R_2 \to \Sigma^{-1} C_1$. The functor then sends the diagram to $\tau_{\leq 1} C_2$.
There is a natural map $C_2 \to M$ given by lifting the augmentation $\epsilon$ along the boundary maps $R_0 = C_0 \to C_1 \to C_2$ — the obstructions to lifting are $\epsilon d_1$ and $\langle \epsilon, d_1, d_2 \rangle$, both of which vanish by being a chain complex.

To show that $\tau_{<1} C_2 \to M$ is an equivalence, we have to show that $C_2 \to M$ induces an isomorphism on the first two homotopy groups.

The filtration of $C_2$ by $C_1$ and $C_0$ gives a spectral sequence for $C_2$ with

$$E^1_{p,q} = \pi_p R_q \Rightarrow \pi_{p+q} \text{Tot}(R_*), \quad d^1 = \pi_* d_q.$$ 

So the $E^2$ page takes the form

$$\begin{array}{c|c}
\text{coker}(\pi_1 d_1) & 0 \\
\text{coker}(\pi_0 d_1) & 0
\end{array} \quad \begin{array}{c}
\text{ker}(\pi_1 d_2) \\
\text{ker}(\pi_0 d_2)
\end{array}$$

The $d^2$ goes from $\ker(\pi_0 d_2)$ to $\text{coker}(\pi_1 d_1)$. This can be described as follows — given $x \in \ker \pi_0 d_2$, we know that $j_2 x$ is in the kernel of $i_1$. So it lifts along $\Sigma^{-1} C_0 \to \Sigma^{-1} C_1$, and the lift is the value of $d_2$. If $x = d_3 \tilde{x}$, then this is exactly the definition of $\langle d_1, d_2, d_3 \rangle \tilde{x} = 0$. But exactness at $R_2$ means that every element in $\ker \pi_0 d_2$ is in the image of $d_3$. So the differential vanishes and

$$\pi_\ast \tau_{\leq 1} C_2 = \text{coker}(\pi_1 d_1) = \pi_\ast M. \quad \square$$

**Remark 1.21.** The “correct” way to reconstruct $M$ is to inductively construct $C_k$ in a similar fashion and take the colimit along $C_k \to C_{k+1}$ to get the total chain complex $C$. This admits a similar spectral sequence except the terms in positive $q$ all vanish, so $C$ is itself equivalent to $M$. We then observe that $\tau_{\leq 1} C_2 = \tau_{\leq 1} C = C$ so we only need the first two stages.

**Corollary 1.22.** Let $\bar{M} \in \text{Mod}_{\pi_0 \mathcal{B}}^\Sigma$ and fix a free homotopy chain complex $R_\ast$ lifting a free resolution of $\bar{M}$. Then there is a bijection between secondary refinements of $\bar{M}$ and diagrams $(\dagger)$ such that $\langle d_1, d_2, d_3 \rangle = 0 \in \text{Hom}_{\pi_0 \mathcal{B}}(\pi_0 R_3, \bar{M}[1])$.

**Proof.** The functor in Lemma 1.20 gives us a function from diagrams $(\dagger)$ to secondary refinements. Since every secondary refinement admits a resolution lifting $R_\ast$, the function is surjective. If two diagrams yield the same secondary refinement, then we can produce a lift of the resolutions

$$\begin{array}{c}
M' \leftarrow R_0 \leftarrow R_1 \leftarrow R_2 \\
M \leftarrow R_0 \leftarrow R_1 \leftarrow R_2
\end{array}$$

and the right-hand side of this diagram gives an equivalence between the two diagrams (note that there may be non-trivial homotopies in the squares). \quad \square

**Theorem 1.23.** Let $\bar{M} \in \text{Mod}_{\pi_0 \mathcal{B}}^\Sigma$. Then there is a well-defined obstruction in $\text{Ext}_{\pi_0 \mathcal{B}}^{3,1}(\bar{M}, \bar{M})$ to the existence of a secondary refinement of $M$. If the obstruction vanishes, then the set of refinements is a torsor over $\text{Ext}_{\pi_0 \mathcal{B}}^{2,1}(\bar{M}, \bar{M})$.

**Proof.** By Corollary 1.22, the obstruction to the existence of a secondary refinement is $\langle d_1, d_2, d_3 \rangle \in \text{Hom}_{\pi_0 \mathcal{B}}(\pi_0 R_3, \pi_1 R_0)$. The indeterminacy from $d_1$-multiplies lets us
replace the target with $\bar{M}[1]$, while the indeterminacy from $d_3$ lets us quotient out by maps that factor through $\pi_0d_3$. Finally,

$$\langle d_1, d_2, d_3 \rangle d_4 = d_1 \langle d_2, d_3, d_4 \rangle,$$

so $\langle d_1, d_2, d_3 \rangle d_4$ vanishes in $\bar{M}[1]$ and $\langle d_1, d_2, d_3 \rangle$ is a cocycle.

Now assume the obstruction vanishes. We have to show that the space of diagrams of the form (†) is a torsor over $\Ext^{2,1}_{\pi_0B}(\bar{M}, \bar{M})$.

The set of null-homotopies is a torsor over $\Hom_{\pi_0B}(\pi_0R_2, \pi_0R_0)$. However, the diagram (†) is equivalent if we modify the null-homotopy by something that factors through $\pi_1d_1$ or $\pi_0d_2$, and the modification must be zero after composition with $d_3$ for the Massey product $\langle d_1, d_2, d_3 \rangle$ to continue to vanish. This gives the desired $\Ext$ group.

□

Lemma 1.24. If two secondary refinements differ by $\chi \in \Ext^{2,1}_{\pi_0B}(\bar{M}, \bar{M})$, then the difference between the secondary differentials is equal to multiplication by $\chi$.

Proof. Let $\epsilon_s: \pi_0R_s \rightarrow \pi_1R_{s-2}$ be the difference in the null-homotopies of $d_{s-1}d_s$. Then they satisfy

$$d_{s-2}\epsilon_s = \epsilon_{s-1}d_s,$$

since their difference is the change in the Massey product $\langle d_{s-2}, d_{s-1}, d_s \rangle$, which vanishes for both refinements. So they form a chain map lifting $\chi$.

But the difference in the secondary differentials is given by composition with $\epsilon$, so we are done. □

Corollary 1.25. Let $X, Y$ be finite type spectra such that $H_*X \cong H_*Y$. Let $d_2^X, d_2^Y$ be their Adams $d_2$ differentials. Then there is a $\chi \in \Ext^{2,1}_{A}(H_*X, H_*X)$ such that

$$d_2^Y(x) = d_2^X(x) + \chi \cdot x.$$

Remark 1.26. We believe the obstruction in Theorem 1.23 is the same as the first obstruction in Toda obstruction theory. Indeed, they are defined in extremely similar ways. However, there are technical issues to a direct comparison that are better addressed in a future paper more systematically.

2. Computation of the secondary differential

The goal of this section is to describe the algorithm to compute secondary resolutions, and hence secondary differentials, in more concrete terms. We start with a more explicit description of a pair algebra.

Let $B$ be a pair algebra, and write $A = \pi_0B$ for conciseness (in our case $A$ will be the Steenrod algebra). Then $B$ be can represented by a 2-term chain complex $\partial: D_1 \rightarrow D_0$, which fits in an exact sequence

$$0 \rightarrow A[1] = \pi_1B \xrightarrow{i} D_1 \xrightarrow{\partial} D_0 \xrightarrow{\pi} A \rightarrow 0.$$

We may assume the multiplication is strictly associative. Then $D_0$ is a ring, $D_1$ is a $D_0$-$D_0$-bimodule, and $\partial$ is a bimodule homomorphism such that

$$(\partial x)y = x(\partial y)$$

for all $x, y \in D_1$.

Our presentations of the secondary Steenrod algebra will satisfy the following extra property:
Definition 2.1. A split pair algebra is a pair algebra \( B = (D_1 \xrightarrow{\partial} D_0) \) equipped with a section

\[
u: R_B \equiv \ker \pi \to D_1\]

that is right-linear over \( D_0 \) and left-linear over \( R_B \).

The presence of such a section lets us capture the information in \( D_1 \) more concisely. On the level of abelian groups, the splitting gives an isomorphism

\[
D_1 \cong R_B \oplus A[1],
\]

and \( \partial: D_1 \to D_0 \) is the projection onto the first factor. In fact, this is an isomorphism of right \( D_0 \)-modules, where \( D_0 \) acts on \( A[1] \) via \( \pi: D_0 \to A \).

To record the left \( D_0 \)-module structure, define the function

\[
i(A', r)) = a \cdot u(r) - u(ar),
\]

noting that the right-hand side is in the kernel of \( \partial \) since \( \partial \) is left-linear over \( D_0 \).

Since \( u \) is left-linear over \( R_B \), the function \( A' \) factors through \( \pi \times R_B \), and gives us a map

\[
A: A \times R_B \to A[1].
\]

This measures the failure of \( u \) to be left-linear over \( D_0 \). The left action of \( D_0 \) on \( D_1 \) is then given by

\[
a \cdot (r, b) = (a \pi(a), r) + \pi(ab).
\]

Thus, the triple \((D_0, \pi, A)\) completely determines the pair algebra.

Remark 2.2. One can check that if \( r \in R_B \) is in the center of \( D_0 \), then

\[
A(-, r): A \to \Sigma^{|r|+1}A
\]

is a derivation.

[Bau06, Theorem 4.5.8] shows that in the secondary Steenrod algebra, \( A(-, 2) \) sends \( Sq^n \) to \( Sq^{n-1} \). It is not \textit{a priori} obvious that this is a derivation. It is a mildly interesting fact, attributed to Bob Bruner, that we can use this to deduce all Adem relations from \( Sq^{2n-1} Sq^n = 0 \) via the Leibniz rule. For practical usage of this observation, note further that the square of a derivation is a derivation mod 2.

Fix any set-theoretic section \( \sigma: A \to D_0 \) of \( \pi \).

Notation 2.3. If \( M \) is a pair module, we represent it as a 2-term chain complex \( \partial: M^{(1)} \to M^{(0)} \).

Fix a pair module \( M \) which we seek to resolve.

Notation 2.4. Let \((\bar{R}_s, d)\) be a minimal resolution of \( \pi_0 M \) over \( A \). Then

\[
\text{Ext}_A^{s,t}(\mathbb{F}_2, \mathbb{F}_2) = \text{Hom}_A^t(\bar{R}_s, \mathbb{F}_2).
\]

Let \( \{g_i\} \) be the \( A \)-module generators of \( \bar{R}_* \). Define \( c_i^j \in A \) by

\[
dg_i = c_i^j g_j
\]

with implicit summation over repeated indices.

The formula \( d^2 = 0 \) translates to

Lemma 2.5. \( c_i^j c_j^k = 0 \).
We then obtain a homotopy chain complex

\[ R_0 \xrightarrow{d_1} R_1 \xrightarrow{d_2} R_2 \xrightarrow{d_3} R_3 \xrightarrow{d_4} R_4 \xrightarrow{d_5} \cdots \]

where \( R_s \) is free with the same generators as \( \bar{R}_s \), and \( d_k g_i = \sigma(c_i^k)g_i \).

To lift this to a chain complex, we need to fill in the null-homotopies \( \delta_s \) of \( d_{s-1}d_s \) in

\[
\begin{array}{cccccc}
R_0^{(1)} & \xleftarrow{d_1} & R_1^{(1)} & \xleftarrow{d_2} & R_2^{(1)} & \xleftarrow{d_3} & R_3^{(1)} & \xleftarrow{d_4} & \cdots \\
\downarrow{\partial} & & \downarrow{\partial} & & \downarrow{\partial} & & \downarrow{\partial} & & \\
R_0^{(0)} & \xleftarrow{d_1} & R_1^{(0)} & \xleftarrow{d_2} & R_2^{(0)} & \xleftarrow{d_3} & R_3^{(0)} & \xleftarrow{d_4} & \cdots \\
\end{array}
\]

Explicitly, the \( \delta_s \) have to satisfy

1. \( d_{s-2}\delta_s = \delta_{s-1}d_s \)
2. \( \partial\delta_s = d_{s-1}d_s \)
3. \( \delta_s\partial = d_{s-1}d_s \).

We have a splitting \( R_s^{(1)} = (\bar{R}_s^{(1)})_{R_B} \oplus (R_s^{(1)})_A \) coming from the splitting of \( D_1 \).

Under this splitting, \( d_s \) is diagonal. Since \( \partial \) maps the \( R_B \) component injectively into \( R_s^{(0)} \), the \( R_B \) component of \( \delta \) is fixed. Specifically, it is given by

\[ \delta_s(x) = u(d_{s-1}d_s x). \]

This automatically satisfies the last two conditions.

The \( A \)-component of \( \delta_2 \) is exactly the data needed to specify the secondary refinement. We take this as given for the moment. We then compute the \( A \)-component of \( \delta_s \) by imposing \( d_{s-2}\delta_s = \delta_{s-1}d_s \). This gives

**Definition 2.6.** For any generator \( g_i \), define

\[ B(a, g_i) = A(a, ddg_i) = A(a, \sigma(c_i^j)\sigma(c_i^k))g_k. \]

We then define

\[ C(g_i) = B(c_i^j, g_j) = A(c_i^j, \sigma(c_i^k)\sigma(c_i^l))g_l. \]

**Theorem 2.7 ([BJ11, Theorem 3.2.11]).** Let \( \bar{R}_* \) be a minimal resolution of \( H^*X \).

Let \( \delta_*: \bar{R}_s \to \bar{R}_{s-2} \) be \( A \)-module homomorphisms such that

\[ d\delta_s g = \delta_{s-1} d g + C(g) \]

for every generator \( g \). Then \( \text{Hom}_A(\delta_s, \mathbb{F}_2) : \text{Hom}^t(\bar{R}_{s-2}, \mathbb{F}_2) \to \text{Hom}^{t+1}(\bar{R}_s, \mathbb{F}_2) \) differs from the Adams \( d_2 \) by multiplication by an element in \( \text{Ext}_A^{2,1}(H^*X, H^*X) \).

Here \( \tilde{\delta} \) is the \( A \)-module homomorphism whose value on generators is the \( A \)-component of \( \delta \).

We can solve this lifting properly inductively, and all that remains is to choose \( \delta_2 \) such that the lifting is possible. A choice of \( \delta_2 \) is a choice of secondary refinement of \( H^*X \). Since the spectrum \( X \) exists, it has a secondary cohomology, so such a choice is possible. In certain easy cases, we can choose \( \delta_2 \) arbitrarily.

**Lemma 2.8.** Suppose either

- \( \text{Hom}(\text{Ext}_A^{2,t}(H^*X, \mathbb{F}_2), H^{t-1}X) = 0 \) for all \( t \); or
- \( \text{Hom}(\text{Ext}_A^{3,t}(H^*X, \mathbb{F}_2), H^{t-1}X) = 0 \) for all \( t \).

Then we can choose \( \delta_2 \) arbitrarily.
In practice, we choose $\tilde{\delta}_2 = 0$.

Proof. In the first case, any choice of $\tilde{\delta}_2$ sends generators to the image of $d_1$, so takes values in the image of $d_1$. So the choice of $\tilde{\delta}_2$ does not affect whether we can lift. Since some choice can be lifted, all choices can.

In the second case, we know any map $R_1 \to R_0[1]$ whatsoever lifts along $d_1$, so the lifting must be possible. □

This covers a large number of cases of interest, e.g. $S$, $S/2^n$ and tmf. In general, picking $\tilde{\delta}_2$ is an exercise in linear algebra.

3. The secondary Steenrod algebra

The original description of the secondary Steenrod algebra was given by [BJ11], where $D_0$ is free over $\mathbb{Z}/4$ on $\{\text{Sq}^n : n \geq 1\}$. This is very large and its $A$ function is fairly complicated. In [Nas12], Nassau discovered a smaller presentation of the secondary Steenrod algebra, which is what we use. In this section, we describe this presentation by providing the triple $(D_0, \pi, A)$.

We start with $D_0$, which is in fact a Hopf algebra. As in the ordinary case, its dual admits a nice “geometric” description — it is the Hopf algebra representing power series of the form

$$f(x) = \sum_{k \geq 0} \xi_k x^{2^k} + \sum_{0 \leq k < l} 2\xi_{k,l} x^{2^k + 2^l}$$

under composition mod 4. This gives a natural inclusion $A_* \hookrightarrow (D_0)_*$, whose dual defines our projection $\pi$.

Explicitly, it is given by

$$(D_0)_* = \mathbb{Z}/4[\xi_k, 2\xi_{k,l} \mid 0 \leq k < l, \xi_0 = 1].$$

with the coproduct given by

$$\Delta \xi_n = \sum_{i+j=n} \xi_i^2 \otimes \xi_j + 2 \sum_{0 \leq k < l} \xi_{n-1-k}^2 \xi_{n-1-l} \otimes \xi_{k,l}$$

$$\Delta \xi_{n,m} = \xi_{n,m} \otimes 1 + \sum_{k \geq 0} \xi_{n-1-k}^2 \xi_{m-1-l} \otimes \xi_{k,l} + \sum_{0 \leq k < l} (\xi_{n-1-k}^2 \xi_{m-1-l} + \xi_{m-1-k}^2 \xi_{n-1-l}) \otimes \xi_{k,l}$$

That is, $(D_0)_*$ is the sub-Hopf algebra of $\mathbb{Z}/4[\xi_k, 2\xi_{k,l}]$ generated by the $\xi_k$ and $2\xi_{k,l}$.

We make the following definition, whose indexing differs slightly from Nassau’s:

**Definition 3.1.** Define $\text{Sq}(R)$ and $Y_{k,l}$ to be dual to $\xi^R_k$ and $2\xi_{k,l}$ under the monomial basis. Set $\sigma(\text{Sq}(R)) = \text{Sq}(R)$.

It is easy to check that

**Lemma 3.2.** $Y_{k,l} \text{Sq}(R)$ is dual to $\xi^R_k \xi_{k,l}$. Further,

$$\pi(Y_{k,l}) = 0 \text{ and } \pi(\text{Sq}(R)) = \text{Sq}(R).$$

**Lemma 3.3.**

$$Y_{s,*} Y_{s,*} = 2Y_{s,*} = 0.$$ 

To describe the rest of the multiplication, we let $\nabla : A_* \otimes A \to A$ be the contraction operator. In the Milnor basis, we have

$$\nabla(\xi^R_k, \text{Sq}(S)) = \text{Sq}(S - R),$$

where $\text{Sq}(S - R)$ is zero if any entry is negative.
Lemma 3.4. If $a \in A$, then
\[ aY_{k,l} = \sum_{i,j \geq 0} Y_{k+i,l+j} \gamma(\xi_i^2 \xi_j^2, a), \]
where we set
\[ Y_{k,l} = \begin{cases} Y_{l,k} & k > l, \\ 2 Sq(\Delta_{k+1}) & l = k. \end{cases} \]

Here $\Delta_k$ is the sequence that is 1 in the $\xi_k$ position and 0 elsewhere.

To determine the multiplication between the $Sq(R)$, recall the following definition in the multiplication of $A$ under the Milnor basis:

Definition 3.5. Let $X = (x_{ij})$ be a matrix indexed on the non-negative integers. Define
\[ r_i(X) = \sum_j 2^j x_{ij}, \quad s_j(X) = \sum_i x_{ij}, \quad t_n(X) = \sum_{i+j} x_{ij}. \]

Define
\[ R(X) = (r_1(X), r_2(X), \ldots), \quad S(X) = (s_1(X), \ldots), \quad T(X) = (t_1(X), \ldots). \]

Let
\[ b(X) = \prod t_n! \prod x_{ij}! = \prod_n \left( x_{n_0} \cdots x_{n_n} \right) \in \mathbb{Z}. \]

Theorem 3.6 ([Mil58, Theorem 4b]). We have
\[ Sq(R) Sq(S) = \sum_{R(X)=R, S(X)=S} b(X) Sq(T(X)). \]

Dualizing the secondary coproduct formula gives

Theorem 3.7.
\[ \sigma(Sq(R)) \sigma(Sq(S)) = \sum_{k \geq 0} \sum_{0 \leq m < n} Y_{m+k,n+k} \gamma(\xi_m^2 \xi_n^2, Sq(R)) \gamma(\xi_k^2, Sq(S)) \]
\[ + \sum_{R(X)=R, S(X)=S} b(X) Sq(T(X)). \]

To specify the full pair algebra, we need to specify the $A$ function. Since it is right-linear over $A$, we only have to specify it on $2$ and $Y_{k,l}$.

Definition 3.8.
\[ A(a, 2) = \gamma(\xi_1, a), \]
\[ A(a, Y_{k,l}) = \sum_{i,j \geq 0} Z_{k+i,l+j} \gamma(\xi_i^2 \xi_j^2, a), \]
where
\[ Z_{k,l} = \begin{cases} 0 & k < l, \\ Sq(\Delta_k + \Delta_{\ell}) & k \geq l. \end{cases} \]
4. Implementation details

The Baues–Jibladze algorithm translates to code pretty directly. In this section, we document some of the less obvious design choices. None of these are novel. However, collecting them in one place could be helpful to future implementors.

4.1. Resolving up to a fixed stem. We first recall the algorithm to compute the minimal resolution of a Steenrod module \( \tilde{M} \) up to a fixed \( t \). Let the resolution be

\[
\tilde{M} \leftarrow d_0 \tilde{R}_0 \leftarrow d_1 \tilde{R}_1 \leftarrow d_2 \tilde{R}_2 \leftarrow d_3 \tilde{R}_3 \leftarrow d_4 \cdots
\]

Let \( R_{s,t} \) be the degree \( t \) component of \( R_s \).

The minimal resolution is constructed inductively by adding generators to bidegree \((s,t)\) so that the chain complex is exact at \((s-1,t)\). Specifically, we perform the following procedure at each bidegree \((s,t)\):

1. Compute the kernel \( K_{s-1,t} \) of \( d_{s-1} \) in degree \( t \).
2. Compute the image \( M_{s-1,t} \) of the existing elements in \( R_{s,t} \) under \( d_{s-1} \). These existing elements come from the generators of degree \( < t \).
3. Find a complement of \( M_{s-1,t} \subseteq K_{s-1,t} \), and add generators to \( R_{s,t} \) to hit this complement.

Naively, it appears that to execute the algorithm at bidegree \((s,t)\), we need to have computed the bidegree \((s-1,t)\) for the first step and \((s,t-1)\) for the second. However, for the first step, we in fact only need to have computed \((s-1,t-1)\), since the generators we add in degree \( t \) cannot belong to the kernel. This is in the same stem as \((s,t)\). This lets us resolve up to a fixed stem instead of a fixed \( t \).

The same observation applies to many other situations, e.g. lifting chain maps.

4.2. Computing binomial coefficients. In the multiplication algorithm, we have to compute binomial coefficients mod 2 and 4. Our main tool is the following version of Lucas’ theorem.

**Lemma 4.1.** The \( p \)-adic valuation of \( \binom{n}{r} \) is the number of borrows when subtracting \( r \) from \( n \) in base \( p \). Equivalently, it is the number of carries when adding \( r \) to \( (n-r) \).

**Corollary 4.2.** \( \binom{n}{r} \equiv 1 \pmod{2} \) iff \( n \& (n - r) = 0 \).

Thus, computing binomial coefficients mod 2 is extremely fast.

To compute the binomial coefficient mod 4, we first compute its \( p \)-adic valuation. We combine Lemma 4.1 with

**Lemma 4.3.** Let \( \text{popcnt}(m) \) be the number of 1’s in the binary representation of \( m \). Then the number of carries when adding \( m \) to \( n \) is \( \text{popcnt}(m) + \text{popcnt}(m + n) \).

In sufficiently modern processors \( \text{popcnt} \) is a single CPU instruction. This lets us quickly check whether \( \binom{n}{r} \) is equal to 0 or 2 mod 4. In case it is odd, we use the general recursion formula:

**Lemma 4.4.** Let \( n = 2^k + \ell \). Then

\[
\binom{n}{r} = \binom{\ell}{r} + 2\binom{\ell}{r - 2^k - 1} + \binom{\ell}{r - 2^k}
\]
The middle term only requires a mod 2 binomial coefficient, which is extremely fast and non-recursive. Moreover, if \( \ell < 2^k \), which can always be arranged, then the first and last terms cannot be simultaneously present. So the recursion can only call binomial-mod-4 once.

**Proof.**

\[
(x + y)^{2^k} \equiv x^{2^k} + 2x^{2^{k-1}}y^{2^{k-1}} + y^{2^k} \pmod{4}. \]

In practice, we use a precomputed table for small values of \( n \). Since the recursion relation above approximately halves \( n \), we can get to the precomputed range pretty quickly.

### 4.3. Products under the Milnor basis

Recall that to compute the product \( \text{Sq}(R) \text{Sq}(S) \) in both the ordinary and secondary Steenrod algebra, we had to iterate through matrices \( X \) such that

\[
\begin{align*}
  r_i &= \sum_j 2^j x_{ij}, \\
  s_j &= \sum_i x_{ij}, \\
  b(X) &= \prod_n \left( \sum_{x_{n0} \cdots x_{0n}} t_n \right) \neq 0.
\end{align*}
\]

We call a matrix **admissible** if it satisfies the first two conditions.

We first lay out some conventions with regards to the matrix. The **inner matrix** is the sub-matrix consisting of terms \( x_{ij} \) with \( i, j \geq 1 \). These entries are called the inner entries.

We always draw the matrix with \( x_{00} \) at the top left, and a diagonal will always refer to a diagonal with slope 1. Such a diagonal would be of the form \( \{ x_{ij} \}_{i+j=n} \), whose entries sum to \( t_n \).

We order the entries of inner matrix \( (x_{ij})_{i,j \geq 1} \) lexicographically. We say

\[
(i, j) < (i', j') \iff i < i' \text{ or } (i = i' \text{ and } j < j').
\]

In other words, we go through the matrix row by row from top to bottom.

The basic strategy is to loop through all admissible matrices, and then compute \( b(X) \) for each of them. We begin with the matrix

\[
x_{i0} = r_i, \quad x_{0j} = s_j, \quad x_{ij} = 0 \text{ for } i, j \geq 1.
\]

We then perform the following procedure:

1. Under the lexicographic order, we find the first inner entry \( x_{ij} \) that can be incremented, i.e.

\[
\sum_{0 \leq k < j} x_{ik} 2^k \geq 2^j, \quad \sum_{0 \leq \ell < i} x_{\ell j} > 0.
\]

2. Increment \( x_{ij} \) by 1, and set every inner entry before it (in the lexicographic order) to 0. Update the \( i = 0 \) and \( j = 0 \) column and row accordingly to remain admissible.

This would then iterate through all admissible matrices in reverse lexicographic order of the inner matrix.

One further optimization is employed. Suppose we have discovered that we can increment \( x_{ij} \). Let \( n \) be the sum of entries in the diagonal to the bottom left of \( x_{ij} \), exclusive. Using the recursive formula for the multinomial coefficient, we know that if we increment \( x_{ij} \) to, say, \( m \), then the multinomial coefficient of this matrix is always zero if \( \binom{n+m}{n} = 0 \). Thus, instead of incrementing \( x_{ij} \) by 1, we increment it to the smallest \( m > x_{ij} \) such that \( \binom{n+m}{n} \neq 0 \).
In the $p = 2$ case, there is a fast way to find this $m$, which I learnt from Bertram Felgenhauer (and is well-known in certain communities):

**Lemma 4.5.** Let $n$, $k$ be such that $n \& k == 0$. The next integer $m > n$ such that $m \& k == 0$ is $(\ell k \mid n) + 1) \& !k$.

The hypothesis that $n \& k == 0$ is always satisfied in our (optimized) algorithm by induction.

4.4. **Memoization.** While the size of the Steenrod algebra grows very quickly (e.g. it has 1189 elements in the $112^{th}$ degree), the number of $Y_{∗,∗}$'s is much smaller, as $Y_{k,ℓ}$ has degree $2^k + 2^ℓ - 1$. Thus, it is relatively cheap to memoize the function $A(a, Y_{k,ℓ})$ and we choose to do so.

4.5. **Performance benchmarks.** We ran the program on a computational server by the Harvard Mathematics Department. It has two Xeon E5-2690 v2 CPUs (10 cores/20 threads each) and 125 GiB of memory.

In Table 1, we document the run time and memory usage of different runs. Details of the runs are as follows:

- We compute $d_2$’s for the sphere up to the specified maximum stem, with the maximum filtration set to half the maximum stem; there are no possible $d_2$’s beyond that.
- We use a pre-computed minimal resolution, and the resources used to compute the minimal resolution is not included.
- We use 39 threads out of the 40 available.

| Stem | Time   | Memory  | Resolution size |
|------|--------|---------|-----------------|
| 40   | 42.0 ms| 13.8 MiB| 1.27 MiB        |
| 50   | 242 ms | 35.4 MiB| 4.60 MiB        |
| 60   | 880 ms | 72.2 MiB| 16.0 MiB        |
| 70   | 8.29 s | 192 MiB | 52.5 MiB        |
| 80   | 53.3 s | 428 MiB | 160 MiB         |
| 90   | 3.45 min| 969 MiB| 462 MiB         |
| 100  | 16.9 min| 2497 MiB| 1254 MiB       |

The computation up to the $140^{th}$ stem was not run continuously and used different versions of the code. We expect that running it again from scratch would take no more than 5 days and 44 GiB of memory.

4.6. **CPU bottlenecks.** With a pre-computed minimal resolution, the majority of the time is spent running the multiplication algorithm of the Steenrod algebra under the Milnor basis. One of the difficulties in optimizing this is that the innermost loop is actually quite fast; any “clever” algorithm to skip invalid configurations may end up being slower due to the increased complexity. We just have to compute lots of products.

While the mod 4 version itself runs much more slowly than the mod 2 version, we also run it much less often and it turns out to be mostly okay. There are more
complex versions of Lemma 4.5 applicable to the mod 4 version, but the runtime performance is mostly indistinguishable from the naive loop.

Another (less significant) bottleneck is in looking up Milnor basis elements. To do linear algebra, we have to pick an ordering of the basis elements in each degree. Getting from the index to a $\text{Sq}(R)$ representation is just looking up an array, which is pretty fast, but going in the other direction is a hashmap lookup which is much slower. This is a potential avenue for improvement — a fast encoding of the basis elements would let us eliminate the hashmap.

4.7. Memory usage. Truth be told, little attention was paid to memory usage; our computer has plenty and it has been feasible to keep everything in memory. If memory is limited, there are obvious avenues for improvement.

When computing to large stems, the majority of memory is used to store the partial inverses along the differentials. For example, the resolution up to the $140^{th}$ stem requires 40 GiB to store, of which 25.8 MiB is used for the values of the differential and the rest are the partial inverses. Since the data is accessed in a very predictable order, storing them on disk should have minimal performance impact.

Part 2. Computing hidden extensions and differentials

We illustrate our strategy with an example. Consider the chart

```
\begin{center}
\begin{tikzpicture}
  \node at (0,0) (a) {a};
  \node at (1,1) (b) {b};
  \node at (2,2) (c) {c};
  \node at (3,3) (d) {d};
  \node at (4,4) (e) {e};
  \node at (5,5) (f) {f};
  \node at (1,1) (g) {g};
  \node at (2,2) (h) {h};
  \node at (3,3) (i) {i};
  \node at (4,4) (j) {j};
  \node at (5,5) (k) {k};
  \node at (0,0) (l) {l};
  \node at (1,1) (m) {m};
  \node at (2,2) (n) {n};
  \node at (3,3) (o) {o};
  \node at (4,4) (p) {p};
  \node at (5,5) (q) {q};
  \node at (0,0) (r) {r};
  \node at (1,1) (s) {s};
  \node at (2,2) (t) {t};
  \node at (3,3) (u) {u};
  \node at (4,4) (v) {v};
  \node at (5,5) (w) {w};
  \node at (0,0) (x) {x};
  \node at (1,1) (y) {y};
  \node at (2,2) (z) {z};
  \node at (3,3) (h1z) {h_1z};
  \node at (4,4) (h1c) {h_1c};
  \node at (5,5) (h1) {h_1};
  \node at (0,0) (h2) {h_2};
  \node at (1,1) (h0) {h_0};
  \node at (2,2) (d0) {d_0};
  \node at (3,3) (d1) {d_1};
  \node at (4,4) (d2) {d_2};
  \node at (5,5) (d3) {d_3};

  \draw (a) -- (b);
  \draw (b) -- (c);
  \draw (c) -- (d);
  \draw (d) -- (e);
  \draw (e) -- (f);
  \draw (f) -- (g);
  \draw (g) -- (h);
  \draw (h) -- (i);
  \draw (i) -- (j);
  \draw (j) -- (k);
  \draw (k) -- (l);
  \draw (l) -- (m);
  \draw (m) -- (n);
  \draw (n) -- (o);
  \draw (o) -- (p);
  \draw (p) -- (q);
  \draw (q) -- (r);
  \draw (r) -- (s);
  \draw (s) -- (t);
  \draw (t) -- (u);
  \draw (u) -- (v);
  \draw (v) -- (w);
  \draw (w) -- (x);
  \draw (x) -- (y);
  \draw (y) -- (z);
  \draw (z) -- (h1z);
  \draw (h1z) -- (h1c);
  \draw (h1c) -- (h1);

\end{tikzpicture}
\end{center}
```

To compute the hidden $\eta$ extension from $x$ to $y$, we look at the Adams spectral sequence for $C\eta$. We will show that if $\tilde{x}$ is a lift of $x$ along $C\eta \to \mathbb{S}$, then there is a hidden $\eta$-extension from $x$ to $y$ iff there is a differential from $\tilde{x}$ to $y$ in $C\eta$.

Once we have computed this hidden extension, the Leibniz rule lets us deduce that $d_2(h_1z) = \eta x = y$. Similarly, a hidden extension from $a$ to $b$ lets us deduce $d_2(b) = h_1c$.

Conventions

- We work in the category of $HF_2$-synthetic spectra throughout.
- We omit all bidegree shifts of synthetic spectra.
- We let $s$ be the Adams filtration, $n$ be the stem, and $t = s + n$. The bidegree of a class will mean the pair $(n, s)$.
- Our code produces a preferred basis of $\text{Ext}$. We will use this basis throughout and write $x_{n,s,i}$ for the $i$th basis element in bidegree $(n, s)$.
- Any other name refer to names used in [IWX20b; IWX20a]. In particular, this includes names of the form $x_{n,s}$ with only two subscripts.
- In all cases, we can sufficiently identify these classes in terms of our basis using only $h_0$, $h_1$, $h_2$ and $d_0$ multiplication. These arguments are documented in Table 7.
- If $x$ is a class on the $E_2$ page, we say it survives to the $E_k$ page if $d_2(x) = ... = d_{k-1}(x) = 0$. We do not require $x$ to be non-zero on the $E_k$ page.
• We frequently make references to [IWX20b], where $\tau$ means the $\tau$ from motivic homotopy theory (i.e. $\tau^2$ in $BP$-synthetic spectra). We will always use $\tau$ for $\tau$ in $HF_2$-synthetic spectra, and $\tau_m$ for what [IWX20b] calls $\tau$.

5. DIFFERENTIALS AND HIDDEN EXTENSIONS

We begin with a quick recap of synthetic spectra, whose relationship with the Adams spectral sequence is laid out in [BHS19, Appendix A]. The main takeaway from the appendix is that the Adams spectral sequence of a spectrum $X$ is “the same as” the $\tau$-Bockstein spectral sequence for $\nu X$. Specifically, they have the same $E_2$ page and their differentials equal up to a sign.

Since the $\tau$-Bockstein is a general categorical construction with good and well-understood properties, we always take “Adams spectral sequence” to mean “$\tau$-Bockstein spectral sequence with the change of sign”, and thus any synthetic spectrum admits an Adams spectral sequence. In this section, $X$ and $Y$ will denote generic synthetic spectra.

Notation 5.1. Let $r_m: X \to X/\tau^m$ be the reduction map, and $\delta_m: X/\tau^m \to X$ the cofiber. Set $\delta = \delta_1$.

If there is no risk of confusion, we also write $r_m: X/\tau^m \to X/\tau^m$ for the reduction map between different cofibers. The cofiber of this map is $r_{n-m}\delta_m$, which we also call $\delta_m$.

Notation 5.2. If $x \in X/\tau^m$ and $y \in X/\tau^n$ are such that $r_kx = r_ky$, we say $x \equiv y \mod \tau^k$. Note in particular that $x$ and $y$ may live in different groups.

Remark 5.3. $\tau^m$ will always denote a map $X/\tau^n \to X/\tau^n$, as opposed to the endomorphism of a $X/\tau^n$ with the same name. In particular, $\tau^m$ is non-zero on $X/\tau$.

We begin with some standard properties of the $\tau$-Bockstein spectral sequence, whose proof is left to the reader.

Lemma 5.4. Let $x \in \pi_{*,*}X/\tau$.

1. For any representative of $d_{k+1}(x)$ on the $E_2$ page, there is a lift of $x$ to $[x] \in \pi_{*,*}X/\tau^k$ such that $\delta_k[x] \equiv -d_{k+1}(x) \mod \tau$.
2. If $\tau^kx = 0$, then $x$ is the target of a $d_{k+1}$ differential.
3. If $\delta x = \tau^{k-2}y$ for some $y$, then $x$ survives to the $E_k$ page, and $d_k(x) \equiv y \mod \tau$.

Lemma 5.5. For any $n, k > m$, we have a commutative diagram

\[
\begin{array}{cccccc}
X & \xrightarrow{\tau^n} & X & \xrightarrow{r_n} & X/\tau^n & \xrightarrow{\delta_n} & X \\
\downarrow{\tau^{n-m}} & & \downarrow{r_m} & & \downarrow{\tau^{n-m}} & & \\
X & \xrightarrow{\tau^m} & X & \xrightarrow{r_m} & X/\tau^m & \xrightarrow{\delta_m} & X \\
\downarrow{\tau^{k-m}} & & & & \downarrow{\tau^{k-m}} & & \\
X & \xrightarrow{\tau^k} & X & \xrightarrow{r_k} & X/\tau^k & \xrightarrow{\delta_k} & X.
\end{array}
\]

We now define hidden extensions. Classically, this is defined for classes on the $E_\infty$ page in terms of multiplication in homotopy groups. For our purposes, we need to generalize this to potentially non-surviving classes. Such a notion was
first introduced by Cooley in his thesis [Coo79, pp. 18–21], together with a version of Theorem 5.10 [Coo79, Theorem 1.24]. Our proposed definition uses synthetic spectra, which we find more ergonomic.

Fix a map of synthetic spectra \( \alpha: X \to Y \). Define \( C\alpha, \iota_\alpha, \delta_\alpha \) by the cofiber sequence

\[
\begin{array}{c}
X \xrightarrow{\alpha} Y \xrightarrow{\iota_\alpha} C\alpha \xrightarrow{\delta_\alpha} X.
\end{array}
\]

**Definition 5.6.** Let \( x \in \pi_{\ast \ast} X/\tau \) and \( y \in \pi_{\ast \ast} Y/\tau \). Suppose \( x \) survives to the \( E_r \) page and \( s < r - 1 \). We say there is a hidden \( \alpha \)-extension by \( s \) from \( x \) to \( y \) on the \( E_r \) page if there is a lift \( [x] \) of \( x \) to \( \pi_{\ast \ast} X/\tau^{r-1} \) and \( [y] \) of \( y \) to \( \pi_{\ast \ast} Y/\tau^{r-1-s} \) such that

\[
\alpha[x] = \tau^s[y].
\]

Alternatively, this says \( \alpha[x] \) is \( \tau^s \) divisible, and a \( \tau^s \) division of \( \alpha[x] \) is equal to \( y \) mod \( \tau \).

We say this hidden extension is maximal if \( \alpha[x] \) is not \( \tau^{s+1} \) divisible. In case \( r = \infty \) and \( \alpha[x] \) is \( \tau^s \) divisible for all \( s \), (e.g. it is zero), we say there is a maximal hidden extension by \( \infty \) to \( 0 \). This is automatic if \( r = s + 2 \).

In particular, a hidden extension by \( 0 \) is a regular, non-hidden extension.

**Remark 5.7.** The jump \( s \) is redundant information given \( x \), \( y \) and \( \alpha \), and we omit it when no confusion can arise.

**Remark 5.8.** After fixing an \( [x] \), the value of \( y \) is well-defined up to images of \( d_2, \ldots, d_{s+1} \), and we shall consider \( y \) as an element in this quotient. It is, however, inaccurate to say it is well-defined on the \( E_{s+2} \) page; it may not survive that long.

Of course, different lifts \( [x] \) give different values of \( y \), and in general they can belong to different filtrations. However, this is not an issue when \( s = 1 \); there is a hidden extension by 1 iff \( \alpha x = 0 \) on the \( E_2 \) page, and the indeterminacy in \( y \) is exactly \( \alpha \)-multiples of classes in the bidegree right above \( x \) on the \( E_2 \) page.

It follows immediately from definition that

**Lemma 5.9.** Suppose \( q < r \) and there is a hidden \( \alpha \)-extension by \( s \) from \( x \) to \( y \) on the \( E_r \) page.

- If \( s + 1 < q \), then there is a hidden \( \alpha \)-extension from \( x \) to \( y \) on the \( E_q \) page.
- If \( s + 1 \geq q \), then there is a hidden \( \alpha \)-extension from \( x \) to 0 on the \( E_q \) page.

**Theorem 5.10.** Let \( x \in \pi_{\ast \ast} X/\tau \) be such that \( d_{k+1}(x) = 0 \). Suppose \( \bar{x} \in \pi_{\ast \ast} C\alpha/\tau \) is such that \( \delta_{\alpha} \bar{x} = x \), and suppose \( y \in \pi_{\ast \ast} Y/\tau \) is such that \( \iota_{\alpha} y = d_{k+1} \bar{x} \) on the \( E_k \) page. Then there is a hidden \( \alpha \)-extension from \( x \) to \( y \) on the \( E_{k+2} \) page.

**Proof.** Consider the cofiber sequences

\[
\begin{array}{cccc}
X & \xrightarrow{\alpha} & Y & \xrightarrow{\iota_\alpha} & C\alpha & \xrightarrow{\delta_\alpha} & X \\
S/\tau & \xrightarrow{\tau_k} & S/\tau^{k+1} & \xrightarrow{\tau_k} & S/\tau^k & \xrightarrow{\delta_k} & X/\tau
\end{array}
\]
Taking the tensor product of these cofiber sequences gives

\[
\begin{array}{ccc}
Y/τ^{k+1} & \xrightarrow{\iota_{a}} & Cα/τ^{k+1} \\
\downarrow{r_k} & & \downarrow{r_k} \\
Y/τ^k & \xrightarrow{\iota_{a}} & Cα/τ^k \\
\downarrow{δ_k} & & \downarrow{δ_k} \\
Y/τ & \xrightarrow{\iota_{a}} & Cα/τ \\
\end{array}
\] \xrightarrow{δ_k} X/τ^{k+1} \xrightarrow{δ_k} X/τ^k \xrightarrow{δ_k} X/τ

Pick a lift \([\bar{x}] \in π_∗Cα \otimes X/τ^k \) of \(\bar{x}\) such that

\[
δ_k[\bar{x}] \equiv -r_αy = -d_{k+1}\bar{x} \mod τ.
\]

By [May01, Section 6] (see also [AM17, Lemma 9.3.2]), there is an \([x] \in X/τ^{k+1}\) such that

\[
r_k[x] = δ_α[\bar{x}], \ α[x] = τ^k y.
\]

The first condition tells us

\[
[x] \equiv δ_α[\bar{x}] \equiv δ_α\bar{x} = x \mod τ.
\]

So \([x]\) is a lift of \(x\) to \(X/τ^{k+1}\), and the result follows.

Since we are able to compute all \(d_2\) differentials, we can compute all \(2, η, \nu\) and \(σ\) hidden extensions by 1 where the target of the extension is in stem at most 111 (since we need \(d_2\) on one stem higher). We have documented a selection of these extensions in Tables 3 to 6, which we use in the coming section.

**Theorem 5.11** (Generalized Leibniz rule). Let \(x ∈ π_∗X/τ\) survive to the \(E_∞\) page. Fix a representative of \(d_r(x)\) on the \(E_2\) page. Then there is a differential from a maximal \(α\)-extension of \(x\) on the \(E_r\) page to a maximal \(α\)-extension of \(d_r(x)\) on the \(E_∞\) page.

**Proof.** Pick a lift \([\bar{x}]\) of \(x\) to \(π_∗X/τ^{r−1}\) such that \(δ_{r−1}[x]\) is a lift of \(-d_r(x)\). Then we have

\[
α[x] = τ^2y, \ αδ_{r−1}[x] = τ^iz
\]

for some \(y\) and \(z\), which are maximal hidden \(α\)-extensions of \(x\) and \(-d_r(x)\) respectively. Then

\[
δy = δ_{r+1}τ^2y = δ_{r+1}α[x] = αδ_{r+1}[x] = τ^{r−2}αδ_{r−1}[x] = τ^{r+1}−2−2z.
\]

This is mostly only useful when the indeterminacy is well-understood, which is always the case when \(s = 0\) or 1. In practice, we use this in combination with Theorem 5.10 and Lemma 5.9. The former lets us compute hidden extensions on the \(E_{s+2}\) page, and the latter tells us any hidden extension on higher pages must be one of those we have already found.

**Remark 5.12.** There are also differentials between the non-maximal extensions, but these all vanish since they are pre-empted by shorter differentials.
6. Computation of new differentials

All new differentials are documented in Table 2, and the proof is indicated in the last column. To avoid circular arguments, when a differential is obtained via division or multiplication, the “original” differential and its justification is also included. These are shown in grey to avoid confusion. This section documents the ones that require a bit more work.

Lemma 6.1. \( d_3(h^3_0 h_6) = \Delta^2 h_0 h^3_3 \).

Proof. By [IWX20b, Theorem 5.7], it is equal to either \( \Delta^2 h_0 h^3_3 \) or \( \Delta^2 h_0 h^2_3 + h_0 M e_0 \). The former is characterized by being in the kernel of \( h \). These are shown in grey to avoid confusion. This section documents the ones that require a bit more work.

Lemma 6.2. \( d_3(h_0 g_4) = h_0 \Delta^2 d_6 e_0 + \Delta h_1 e^2_0 g \).

Proof. First, \( h_0 g_4 \) is \( h_2 \)-divisible with

\[ h_0 g_4 = h_2 e_0 B_4, \quad d_2(e_0 B_4) = h_0 M d_0 e_0. \]

There is a hidden \( \nu \)-extension from \( h_0 M d_0 e_0 \) to \( \Delta h_1 e^2_0 g \) with indeterminacy \( h_0 \Delta^2 d_6 e_0 \). So we know that

\[ d_3(h_0 g_4) = \Delta h_1 e^2_0 g + h_0 \Delta^2 d_6 e_0. \]

To determine the indeterminacy, we consider further \( \nu \)-multiplication. There is a hidden \( \nu \)-extension from \( h_0 g_4 \) to \( h_0 \Delta^2 m \). Thus, we find that

\[ h_2 d_3(h_0 g_4) = d_2(h_0 \Delta^2 m) = h_2(h_0 \Delta^2 d_6 e_0). \]

Since \( h_2 \Delta h_1 e^2_0 g = 0 \), the result follows.

Remark 6.3. Our calculations have discovered two incorrect \( d_3 \)'s in [IWX20b].

[IWX20b, Lemma 5.21] claims that \( d_3(h_0 g_4) = \Delta^3 h_0 d_0 e \) (note that \( d_0 B_5 = h_0 g_4 \) in the classical Adams spectral sequence). Their argument neglects the possibility that in the motivic Adams spectral sequence, \( d_3(\tau^m_3 d_0 B_5) = \Delta^2 h_0 d_0 e_0 + \tau^m_3 \Delta h_1 e^2_0 g \), which would make \( \Delta^2 h_0 d_0 e_0 \) \( \tau^m_3 \)-divisible in the \( E_\infty \)-page. Indeed, our argument shows this is exactly what happens.

[IWX20b, Lemma 5.26] claims that \( d_3(M h_0 d_0 k) = P \Delta^2 h_0 d_0 e_0 \). This is a clerical error; in mmf, there is a \( d_2 \) hitting \( \Delta h_1 d^2_0 e^2_0 + \tau^3_3 P \Delta h_1 d g^2 \), so in the \( E_3 \) page we have \( \Delta h_1 d^2_0 e^2_0 = \tau^3_3 P \Delta h_1 d g^2 \). Since \( \tau^2_3 h_0 M d_0 k \) has trivial image in mmf, its \( d_3 \) must be \( \Delta h_1 d^2_0 e^2_0 + \tau^3_3 P \Delta h_1 d g^2 \).

Lemma 6.4. \( d_3(\Delta^3 h_1 h_3) = \Delta h_1 e^2_0 g \).

Proof. We first show that \( d_3(\Delta^3 h_1 h_3) \) is non-zero and not equal to \( d_3(h_0 g_4) \). This is the argument of [IWX20b, Lemma 5.20], after correcting for Remark 6.3. Since \( Ph_1 \cdot \Delta^3 h_1 h_3 = Ph_1(\Delta^3 h_1 h_3 + h_0 g_4) \) supports a \( d_4 \), both \( \Delta^3 h_1 h_3 \) and \( \Delta^3 h_1 h_3 + h_0 g_4 \) support differentials of length at most \( d_4 \). The target bidegree of the \( d_4 \) is zero and computer calculations show they do not support \( d_2 \)'s, so they must support a \( d_3 \).

This implies \( d_3(\Delta^3 h_1 h_3) = h_0 \Delta^2 d_0 e_0 \) or \( \Delta h_1 e^2_0 g \). However, we notice that \( \Delta^3 h_1 h_3 \) has a hidden \( \nu \)-extension by 1 to 0, so \( h_2 d_3(\Delta^3 h_1 h_3) = 0 \). So it is equal to \( \Delta h_1 e^2_0 g \).

Lemma 6.5. \( d_6(h_1 \Delta^2 g_2) = 0 \).
Proof. We use that $h_1 \Delta^2 g_2 = h_3 \Delta^2 e_1$. Since $d_3(\Delta^2 e_1) = h_3^2 \Delta^2 n$, we know that
\[ \delta(\Delta^2 e_1) = \tau \nu [\Delta^2 n] + \tau^2 \eta [M \Delta h_1 d_0] + \tau^3 [\Delta h_1 g^3]. \]
Since hidden $\sigma$-extensions by 1 vanish identically at bidegree $(85, 17)$, we know that

\[ \sigma[\Delta h_1 g^3] = 0 \mod \tau^2. \]

Moreover, computer calculation tells us $\sigma[M \Delta h_1 d_0] = 0 \mod \tau^2$, and $h_2 M d_0 k$ supports a differential. So $\sigma[M \Delta h_1 d_0] = 0 \mod \tau^3$, and

\[ \delta(h_3 \Delta^2 e_1) = \sigma \delta(\Delta^2 e_1) \equiv 0 \mod \tau^5. \qed \]

\textbf{Lemma 6.6.} $d_3(\log_{94,8}) = h_1 x_{92,10}$.

\textit{Proof.} This follows from [IWX20b, Remark 5.2] since $d_2(\log_{94,8}) = 0$. \qed

\section{Tables}

This section contains the following tables:

- Table 2 contains all the newly computed differentials and their proofs. The grey rows consist of old differentials we include for reference.
- Tables 3 to 6 contains the hidden extensions that we use to compute the differentials. The first four columns are lifted straight out of computer-generated data, while the last two columns identify the names of the classes.
- Table 7 gives the identification between the names in [IWX20b] and our basis, together with a brief justification for each. We omit cases where the group is one-dimensional.

\begin{table}[h]
\centering
\begin{tabular}{|cccccc|}
\hline
$n$ & $s$ & $r$ & source & target & proof \\
\hline
63 & 8 & 3 & $h_0^2 h_6$ & $\Delta^2 h_0 h_3^2$ & Lemma 6.1 \\
69 & 8 & 2 & $D_3'$ & 0 & - \\
69 & 8 & 3 & $D_3'$ & $h_2 M g$ & 2 division \\
69 & 10 & 2 & $P(A + A')$ & $h_0 h_2 M g$ & - \\
80 & 14 & 3 & $h_0 g B_3$ & $h_0 \Delta^2 d_0 e_0 + \Delta h_1 e_0^2 g$ & Lemma 6.2 \\
80 & 14 & 3 & $\Delta^3 h_1 h_3$ & $\Delta h_1 e_0^2 g$ & Lemma 6.4 \\
85 & 17 & 2 & $M d_0 j$ & $h_0 M P d_0 e_0$ & - \\
87 & 17 & 2 & $\Delta^3 h_1 d_0$ & $e_0^2 e_0^2 m$ & - \\
88 & 18 & 3 & $\Delta^3 h_1 d_0$ & $\Delta h_1 d_0^2 e_0^2$ & $\eta$ multiplication \\
88 & 18 & 3 & $h_2 M d_0 j$ & $\Delta h_1 d_0^2 e_0^2 + h_0 P \Delta^2 d_0 e_0$ & $\nu$ multiplication \\
92 & 12 & 5 & $\Delta^2 g_2$ & 0 & $\eta$ division \\
93 & 10 & 6 & $h_0^3 \Delta h_2 h_6$ & $M \Delta h_2 e_0$ & 2 multiplication \\
93 & 13 & 6 & $h_1 \Delta^2 g_2$ & 0 & Lemma 6.5 \\
94 & 8 & 2 & $x_{94,8}$ & 0 & - \\
94 & 8 & 3 & $x_{94,8}$ & $h_1 x_{92,10}$ & Lemma 6.6 \\
94 & 15 & 3 & $\Delta^2 M h_1$ & $M d_0 e_0^2$ & 2 division \\
94 & 17 & 3 & $M d_0 m$ & $M P \Delta h_1^2 d_0$ & $\eta$ division \\
95 & 7 & 2 & $x_{95,7}$ & $h_0 x_{94,8}$ & - \\
95 & 16 & 4 & $\Delta^2 M h_1$ & $M P \Delta h_1^2 e_0$ & $\eta$ multiplication \\
95 & 19 & 2 & $x_{95,19,0}$ & $M P \Delta h_1^2 d_0$ & - \\
\hline
\end{tabular}
\caption{Newly computed differentials}
\end{table}
Table 3: Selected hidden 2-extensions

| n  | s  | source | target | name of source | name of target |
|----|----|--------|--------|---------------|---------------|
| 69 | 8  | [1, 0] | [1]    | $D'_3$         | $P(A + A')$    |
| 92 | 14 | [0, 1, 0]| [1]    | $h_0^2 \Delta^2 g_2$ | $M \Delta h_0^2 e_0$ |
| 93 | 18 | [1]    | [1, 0] | $Md_0 e_0^2$    | $M P \Delta h_0^2 d_0$ |
| 94 | 15 | [1]    | [1]    | $\Delta^2 M h_1$ | $Md_0 m$ |

Table 4: Selected hidden $\eta$ extensions

| n  | s  | source | target | name of source | name of target |
|----|----|--------|--------|---------------|---------------|
| 86 | 19 | [1]    | [0, 1, 1]| $e_0^3 m$     | $\Delta h_1 d_0^2 e_0^2$ |
| 91 | 17 | [0, 0, 1]| [0, 1] | $Md_0 l + h_0^2 x_{91,11}$ | $M P \Delta h_1 d_0$ |
| 93 | 18 | [1]    | [0, 1] | $Md_0 e_0^2$    | $M P \Delta h_0^2 e_0$ |
| 94 | 17 | [1]    | [1, 0] | $Md_0 m$       | $x_{95,19,0}$ |

Table 5: Selected hidden $\nu$ extensions

| n  | s  | source | target | name of source | name of target |
|----|----|--------|--------|---------------|---------------|
| 63 | 8  | [1, 0] | [1, 0] | $h_0^2 h_6$   | $B_5 + D'_2$ |
| 76 | 15 | [1]    | [0, 1, 1]| $h_0 M d_0 e_0$ | $\Delta h_1 e_0^2 g$ |
| 80 | 14 | [1, 0] | [0, 1] | $h_0 g B_4$    | $h_0 \Delta^2 m$ |
| 80 | 14 | [0, 1] | [0, 0] | $\Delta^3 h_1 h_3$ | 0 |
| 84 | 19 | [1, 1] | [0, 0, 1]| $h_0 M P d_0 e_0$ | $\Delta h_1 d_0^2 e_0^2 + h_0 P \Delta^2 d_0 e_0$ |

Table 6: Selected hidden $\sigma$ extensions

| n  | s  | source | target | name of source | name of target |
|----|----|--------|--------|---------------|---------------|
| 84 | 15 | [1, 1] | [0, 0, 0]| $M \Delta h_1 d_0$ | 0 |
| 85 | 17 | [1, 0, 0]| [0, 0] | $x_{85,17,0}$ | 0 |
| 85 | 17 | [0, 1, 1]| [0, 0] | $x_{85,17,1} + x_{85,17,2}$ | 0 |

Table 7: Identification of classes

| n  | s  | class | name | identification |
|----|----|-------|------|---------------|
| 62 | 11 | [1, 1] | $\Delta^2 h_0 h_6^2$ | $h_2$-torsion |
| 63 | 8  | [1, 0] | $h_0^2 h_6$ | $h_0$-divisible |
| 66 | 10 | [1, 1] | $B_5$ | $h_0^2$-torsion |
| 66 | 10 | [0, 1]| $D'_2$ | $h_3 D'_2$ is $h_2$-divisible |
| 68 | 11 | [1, 0] | $h_2 M g$ | $h_2$-divisible |
| 68 | 12 | [1, 0] | $h_0 h_2 M g$ | $h_2$-divisible |
| 69 | 8  | [1, 0] | $D'_3$ | $h_0$-torsion |
| 79 | 17 | [0, 1, 1]| $\Delta h_1 e_0^2 g$ | $h_0$-torsion |
### Table 7: Identification of classes (continued)

| n  | s  | class   | name             | identification |
|----|----|---------|------------------|----------------|
| 79 | 17 | [1, 0, 0] | $h_0 \Delta^2 \delta_0 e_0$ | $h_0$-divisible |
| 80 | 14 | [0, 1]   | $\Delta^3 \delta_1 h_3$     | $h_0$-torsion   |
| 80 | 14 | [1, 0]   | $h_{00} B_4$              | $h_0$-divisible |
| 83 | 16 | [0, 1]   | $h_0 \Delta^2 m$          | $h_0$-divisible |
| 84 | 15 | [1, 1]   | $M \Delta h_1 d_0$       | $h_0$-torsion   |
| 84 | 19 | [1, 1]   | $h_0 MP \delta_0 e_0$     | $h_2$-divisible |
| 87 | 21 | [0, 1, 0] | $h_0 P \Delta^2 \delta_0 e_0$ | $h_0$-divisible |
| 87 | 21 | [0, 1, 1] | $\Delta h_1 \delta_0^2 \delta_0^2$ | $h_0$-torsion   |
| 91 | 17 | [1, 0, 0] | $M d_0 l$             | $d_0$-divisible |
| 91 | 17 | [1, 0, 0] | $h_0^6 x_{91, 11}$     | $h_0$-divisible |
| 92 | 14 | [0, 1, 0] | $h_0^3 \Delta^2 g_2$ | $h_0$-divisible |
| 92 | 19 | [0, 1]   | $M P h_1 \delta_0 +? e_0 g^2 m$ | $h_1$-non-torsion |
| 93 | 10 | [1, 1, 0] | $h_0^3 \Delta h_2^2 \delta_0$ | $h_0$-divisible |
| 93 | 11 | [1, 1]   | $h_1 x_{92, 10}$          | $h_1$-divisible |
| 93 | 20 | [1, 0]   | $M P \Delta h_2^2 \delta_0$ | $h_1$-divisible |
| 94 | 20 | [0, 1]   | $M P \Delta h_2^3 \delta_0 +? e_0^3 g$ | $h_0$-non-torsion |

A.1. **Installation and running.** The code used for the calculation is available at [CCB21](https://github.com/SpectralSequences/sseq), and the latest version of this software is available at https://github.com/SpectralSequences/sseq. This is a monorepo, and we will work in the `ext/` subdirectory. This repository comes with a reasonable amount of documentation, and the README in `ext/` contains instructions for accessing said documentation.

To assist the reader in reproducing the results, we describe the full interactive session that generates the data we need. This guide is written for the version in [CCB21] but should work with future versions with little modifications.

Assuming Rust is installed, running the commands as indicated below in any subdirectory of `ext/` will compute the $d_2$ differentials for $S_2$ and $C^2$ as well as the hidden $2$-extensions of the sphere.

We begin by computing the $d_2$ differentials, filtration one products, and a listing of the dimensions of $\text{Ext}$ in each bidegree. The first two are directly useful, while the last is used for computing hidden extensions$^2$.

```bash
$ cargo run --example resolve_through_stem
Module (default: S_2): S_2@milnor
Max s (default: 15): 20
Max n (default: 30): 40
Save file (optional): resolution_S_2.save

$ cargo run --example secondary > d2_S_2
Module (default: S_2): S_2@milnor
Resolution save file (optional): resolution_S_2.save
```

$^2$As mentioned in the README, when resolving to larger stems, one ought to supply the `--release`, `--features concurrent` and `--no-default-features` flags for much improved performance.
We repeat these commands with $S_2$ replaced by $C_2$.

To compute hidden extensions, we need the maps induced by the inclusions and projections $S \rightarrow C_2 \rightarrow S_1$. We use the `lift_hom` script that computes the map $\text{Ext}(N,k) \rightarrow \text{Ext}(M,k)$ induced by an element of $\text{Ext}(M,N)$.

Finally, we run a script to put together these information and obtain the hidden extensions

```bash
$ cargo run --example hidden > hidden_2
Max s: 20
Max n: 40
Stem of $\alpha$: 0
Name of $\alpha$: h_0
Dimension of Ext of $X$: num_gens_S_2
Dimension of Ext of $X/\alpha$: num_gens_C2
$\alpha$ products: filtration_one_S_2
Inclusion map: inclusion_C2
Projection map: projection_C2
```
d2 of X: d2S_2

d2 of X/α: d2C2

These hidden extensions can also be reasonably computed by hand for any bidegree. Indeed we have been doing exactly that for a while before putting the script together.

A.2. Raw data. Raw data printed out by the program is available at [Chu21]. The available data are as follows:

- **d2**: The $d_2$ differentials, generated by `secondary`. If the target bidegree is empty, the differentials are omitted.
- **filtration_one**: All filtration-one products, generated by `filtration`. This is useful for manually identifying classes in different bases.
- **kappa, kappabar**: Products with $d_0$ and $g$, generated by `lift_hom`.
- **hidden_***: Hidden extensions by the corresponding class, generated by `hidden`.
- **inclusion**: The map in Ext induced by inclusion of the bottom cell, generated by `lift_hom`.
- **projection**: The map in Ext induced by projection to the top cell, generated by `lift_hom`.
- **change_of_basis**: The correspondence between our basis and Bruner’s basis, generated by `bruner`.
- ***charts.***: The Adams charts showing the $d_2$’s and the $E_3$ page, in various formats. Then “clean” versions omit the $h_2$ products.
- **module.json**: Files defining the module.
- **num_gens**: Dimension of Ext in each bidegree, generated by `num_gens`.
- **differentials.gz**: The differentials in our minimal resolution, generated by `differentials`. This, in theory, uniquely specifies our basis elements, and can be used to compare our basis with other resolutions.
- **bruner.tar.gz**: The differentials in our resolution in Bruner’s format, generated by `save_bruner`.

See [Chu21] for more details.

### References

[AM17] Michael Andrews and Haynes Miller. “Inverting the Hopf map”. eng. In: *Journal of topology* 10.4 (2017), pp. 1145–1168.

[Bau06] Hans-Joachim Baues. *The Algebra of Secondary Cohomology Operations*. eng. 1st ed. 2006. Progress in Mathematics, 247. Basel: Birkhäuser Basel: Imprint: Birkhäuser, 2006.

[BHS19] Robert Burklund, Jeremy Hahn, and Andrew Senger. *On the boundaries of highly connected, almost closed manifolds*. 2019. arXiv: 1910.14116 [math.AT].

[BJ06] Hans-Joachim Baues and Mamuka Jibladze. “Secondary derived functors and the Adams spectral sequence”. In: *Topology* 45.2 (2006), pp. 295–324.

[BJ11] Hans-Joachim Baues and Mamuka Jibladze. “Dualization of the Hopf algebra of secondary cohomology operations and the Adams spectral sequence”. In: *Journal of K-Theory* 7.2 (2011), pp. 203–347.
[CCB21] Dexter Chua, Hood Chatham, and Joey Beauvais-Feisthauer. SpectralSequences/sseq: v0.0.1. Version v0.0.1. May 2021. doi: 10.5281/zenodo.4766472.

[Chu21] Dexter Chua. Computer calculated Adams $d_2$ differentials. Version v1.0.0. Zenodo, May 2021. doi: 10.5281/zenodo.4766457.

[Coo79] Clifford Robert Cooley. The Extended Power Construction and Cohomotopy. Thesis (Ph.D.)—University of Washington. ProQuest LLC, Ann Arbor, MI, 1979, p. 88.

[IWX20a] Daniel C. Isaksen, Guozhen Wang, and Zhouli Xu. “Classical and motivic Adams charts”. In: (2020). eprint: s.wayne.edu/isaksen/adams-charts.

[IWX20b] Daniel C. Isaksen, Guozhen Wang, and Zhouli Xu. More stable stems. 2020. arXiv: 2001.04511 [math.AT].

[May01] J.P. May. “The Additivity of Traces in Triangulated Categories”. In: Advances in Mathematics 163.1 (2001), pp. 34–73.

[Mil58] John Milnor. “The Steenrod Algebra and Its Dual”. eng. In: Annals of mathematics 67.1 (1958), pp. 150–171.

[Nas12] Christian Nassau. “On the secondary Steenrod algebra”. In: New York J. Math. 18 (2012), pp. 679–705.

[PV19] Piotr Pstrągowski and Paul VanKoughnett. Abstract Goerss-Hopkins theory. 2019. arXiv: 1904.08881 [math.AT].