

1 **baerhunter**: An *R* package for the discovery and analysis of
2 expressed non-coding regions in bacterial RNA-seq data

3

4 A. Ozuna¹, D. Liberto¹, R. M. Joyce¹, K.B. Arnvig², I. Nobeli^{1*}

5 ¹ Institute of Structural and Molecular Biology, Department of Biological
6 Sciences, Birkbeck, Malet Street, London WC1E 7HX, UK

7 ² Institute of Structural and Molecular Biology, University College London,
8 London WC1E 6BT, UK

9

10 * Corresponding author: i.nobeli@bbk.ac.uk

11 **Abstract**

12 **Summary**

13 Standard bioinformatics pipelines for the analysis of bacterial transcriptomic
14 data commonly ignore non-coding but functional elements e.g. small RNAs, long
15 antisense RNAs or untranslated regions (UTRs) of mRNA transcripts. The root of
16 this problem is the use of incomplete genome annotation files. Here, we present
17 *baerhunter*, a method implemented in *R*, that automates the discovery of
18 expressed non-coding RNAs and UTRs from RNA-seq reads mapped to a
19 reference genome. The core algorithm is part of a pipeline that facilitates
20 downstream analysis of both coding and non-coding features. The method is
21 simple, easy to extend and customize and, in limited tests with simulated and
22 real data, compares favourably against the currently most popular alternative.

23 **Availability**

24 The *baerhunter* *R* package is available from:
25 <https://github.com/irilenia/baerhunter>

26 **Contact**

27 i.nobeli@bbk.ac.uk

28 Introduction

29

30 Next-generation sequencing has facilitated global surveys of the transcriptome,
31 largely focused on studying differential expression of genes across different
32 conditions. Studies of eukaryotic transcriptomes are increasingly embracing the
33 analysis of non-coding transcript expression but in bacteria, where intergenic
34 regions tend to be a lot shorter (Thorpe *et al.*, 2017) and less well annotated,
35 automated differential gene expression is still largely synonymous with
36 differential expression of the coding regions (CDS). As the functional importance
37 of bacterial non-coding RNAs (ncRNAs - the term used here to cover long
38 antisense RNA, small regulatory RNA (sRNA), and untranslated parts of mRNAs)
39 is becoming evident (Michaux *et al.*, 2014), so is the need for including them in
40 differential expression studies.

41

42 A major obstacle in studying non-coding RNA expression in bacteria is that
43 relatively few ncRNAs are reliably annotated and, with the exception of well-
44 known cases (such as tRNAs, ribosomal RNAs and, more recently, some members
45 of the RFAM (Kalvari *et al.*, 2018) families), the majority are not included in the
46 standard annotation files required by computational pipelines. Requiring the
47 non-coding RNAs to be included in the annotation is prohibiting their analysis by
48 methods such as TrBorderEx (Wang *et al.*, 2015), which identifies the transcript
49 boundaries but does not find new non-coding RNAs. An alternative to waiting for
50 annotations to improve is to identify ncRNAs using the expression data signal.
51 Early efforts in this direction relied on a combination of manual inspection and
52 in-house written scripts to identify clusters of reads falling outside known CDS
53 regions (Arnvig *et al.*, 2011)(Wilms *et al.*, 2012)(Pfeifer-Sancar *et al.*, 2013).
54 These studies offered great insights into the non-coding transcriptome but
55 applying their approach in a different context is time-consuming and prone to
56 errors due to the need for recreating the pipelines from scratch. Selected studies
57 have led to publicly available software for the study of ncRNAs in bacteria.
58 However, some methods are limited to specific species (Pellin *et al.*, 2012) or
59 rely on specialized sequencing protocols (Peña-Castillo *et al.*, 2015; Amman *et*
60 *al.*, 2014). Two notable exceptions have appeared in recent years. DETR'PROK

61 (Toffano-Nioche *et al.*, 2013) employs the Galaxy platform (Afgan *et al.*, 2018) to
62 classify clusters of RNA-seq reads not overlapping with annotated genes as
63 sRNAs, antisense RNAs, and UTRs. The updated annotation file can be used to
64 carry out differential gene expression. However, the DETR'PROK workflow is
65 composed of a large number of steps, requires an active user input at several
66 stages and depends on access to a Galaxy instance. Rockhopper (McClure *et al.*,
67 2013), a standalone, Java-based program that allows both identification of
68 features in a bacterial transcriptome and differential expression between
69 conditions, is primarily aimed at non-bioinformaticians. A user-friendly
70 graphical interface masks a fairly sophisticated set of algorithms that are
71 presented as a black box with only a handful of parameters accessible to the
72 user. Although straightforward to use, the set up is inflexible with little scope for
73 extending or altering the pipeline without expert interfering with the code.

74

75 Here, we present *baerhunter* (“baer” stands for **b**acterial **e**xpressed **r**egions), a
76 new method implemented in *R* (R Core Team, 2018), for automating the
77 detection and quantification of expressed putative non-coding RNAs (including
78 UTRs) in bacterial strand-specific RNA-seq data. At the core of the *baerhunter*
79 pipeline is a simple but effective method of capturing expressed intergenic
80 regions across sets of RNA-seq data samples. The method is designed to provide
81 predictions of approximate locations of non-coding elements, reflecting our
82 belief that accurate definitions of transcript ends are best achieved by targeted
83 experimental methods rather than computational predictions from noisy data.
84 The pipeline built around this method facilitates the analysis of differential
85 expression of these regions in parallel with the more traditional protein-coding-
86 focused analysis. Below, we describe our method and present the results of
87 testing its performance both on simulated and real data from *Mycobacterium*
88 *tuberculosis* (*Mtb*). In addition, we compare *baerhunter* to Rockhopper, chosen as
89 the most widely used alternative method.

90 **Methods**

91 The core algorithm of *baerhunter* carries out a search for intergenic features on
92 each strand, displaying a minimum length and coverage depth in the RNA-seq
93 signal (see Supplementary Methods for details). The algorithm is wrapped
94 within a “driver” R script that can be easily edited to include, exclude or modify
95 steps, depending on the user’s requirements. Individual functions of *baerhunter*
96 can also be used in isolation or incorporated within different pipelines. In the
97 default mode, *baerhunter* reads in a set of Binary Alignment Map (BAM) files
98 with RNA-seq reads mapped to a reference genome and an annotation file in the
99 Generic Feature Format (GFF3) for the same genome. It will then identify
100 expressed intergenic regions on each strand (“features”) and combine
101 overlapping features across multiple BAM files to create a full set of non-
102 overlapping genomic features. Features are classified as either “UTR”, if they are
103 thought to be the untranslated part of a coding mRNA or “sRNA” (used in this
104 context to encompass all other types of non-coding RNA in bacteria, including
105 long antisense RNAs). In addition, *baerhunter* allows for new transcripts to be
106 filtered by their expression level (normalised to transcripts per million (TPM)
107 values), as many very low-expression features are likely to be the result of
108 transcriptional noise or ambiguous read mapping. Finally, differential expression
109 analysis, including all newly annotated putative features, is facilitated by a
110 wrapper script that utilizes the DESeq2 method (Love *et al.*, 2014).

111

112 To test *baerhunter*, a simulated RNA-seq dataset was created using the package
113 *polyester* (Frazee *et al.*, 2015). In addition, RNA-seq data from the study of
114 (Cortes *et al.*, 2013), six samples from exponentially growing and starved
115 cultures of *Mtb*, were downloaded from Array Express (E-MTAB-1616) and
116 processed as detailed in Supplementary Methods. Following analysis with
117 *baerhunter*, the sRNA/UTR predictions were compared to a set of experimentally
118 confirmed and predicted mycobacterial sRNAs from the comprehensive review
119 of (Haning *et al.*, 2014). Transcription start sites reported by (Cortes *et al.*, 2013)
120 for the same samples were also used to assess the accuracy of our predictions.
121 The genome browser Artemis (version 17.0.1) (Carver *et al.*, 2012) was used for
122 visualization.

123

124 Rockhopper was used with default parameters, except for the minimum
125 transcript length that was set to 40 nucleotides to match the *baerhunter* settings.
126 Two minimum expression thresholds were tested (0.5, the default, and 0.2, to
127 increase sensitivity).

128

129 All code and data required to reproduce this analysis are available from the
130 github repository:

131 https://github.com/irilenia/baerhunter_paper

132 The latest version of *baerhunter* is available from:

133 <https://github.com/irilenia/baerhunter>

134

135

136 **Results**

137 **Simulated dataset**

138 We tested the ability of *baerhunter* to recover expressed intergenic regions and
139 UTRs using simulated data. 1000 genomic features were randomly selected from
140 the *Mtb* genome, including twenty-four short RNAs included in the original
141 annotation (see Supplemental Methods). As the genome annotation file does not
142 include UTR information for *Mtb*, artificial UTRs were added to a random subset
143 of 200 genes. These 1000 features were simulated in 10 samples belonging to
144 two groups (with fold changes between 1 and 5 applied to 20% of the features).
145 Our pipeline applied to paired-end read simulations recovered all short RNAs
146 and all UTRs, with exact predictions for the start and end coordinates of all 24
147 sRNAs and over half of the UTRs (the remaining being in their vast majority
148 within 5 nucleotides of the true range). Results were relatively insensitive to
149 small changes in the program parameters (Supp Table 1). Rockhopper
150 performed similarly on sRNAs, recovering 23 of 24 when run at default
151 sensitivity (22 of the 23 sRNAs had their coordinates exactly predicted) but was
152 less successful in the prediction of UTRs, missing 4 of the 200 and estimating the

153 lengths of approximately a quarter of the ones it predicted to be at least 20
154 nucleotides shorter than expected (Supp Figure 4).

155

156 **Real dataset**

157 In addition to using simulated data, we applied *baerhunter* to the RNA-seq data
158 from starved and exponentially grown cultures of *Mtb* (Cortes *et al.*, 2013). In
159 this case, the true number of non-coding RNAs is unknown, so *baerhunter* was
160 benchmarked against transcription start-site data from the same samples, as
161 well as lists of known and predicted non-coding RNAs (Haning *et al.*, 2014) in
162 order to assess the likely accuracy of our predictions.

163

164 At the more stringent parameter values (5-20), 74-83% of the predicted sRNA
165 features in samples from either condition are supported by the presence of a
166 TSS, even at one-nucleotide resolution (Supp Figure 5A & Supp Table 2).
167 Relaxation of the cut-off (to 5-10) increases false positives but, importantly it
168 also increases true predictions, thus allowing more transcripts to be discovered
169 at the cost of a more noisy output (Supp Fig. 5B & Supp Table 2). Although more
170 than half of the *baerhunter*-predicted sRNAs do not correspond to sRNAs in the
171 published list of (Haning *et al.*, 2014), visual examination of the RNA-seq signal
172 confirms expression at these loci (Supp Figure 6), usually from a very weak TSS
173 that has not passed the inclusion cut-off in the original study by (Cortes *et al.*,
174 2013). These transcripts are often expressed at very low levels and can be easily
175 filtered out using expression strength. Rockhopper, run at default expression
176 cut-offs, not only predicts fewer sRNAs but also a smaller percentage of these
177 predictions (~50-75%) are supported by TSS evidence (Supp Figure 5C&D). The
178 prediction of UTRs is harder to assess. In the absence of ground truth for 3'
179 UTRs, we compared the start of the 5' UTR predictions to TSS data. Rockhopper
180 retrieved more of the 5' UTRs using default sensitivity but differences were
181 generally small (Supp Figure 7). Importantly, the two programs differ in their
182 treatment of 5' UTRs that are followed by a significant drop near the start of the
183 CDS. Rockhopper annotates these as independent sRNAs, a design that
184 occasionally leads to the unintended consequence of predicting a much shorter
185 5' UTR with no experimental support (Figure 1A). In *baerhunter*, we prefer to

186 treat these as 5' UTRs, acknowledging that they originate from mRNAs
187 containing transcription-terminating regulatory elements ahead of the coding
188 region.
189

190 Conclusion

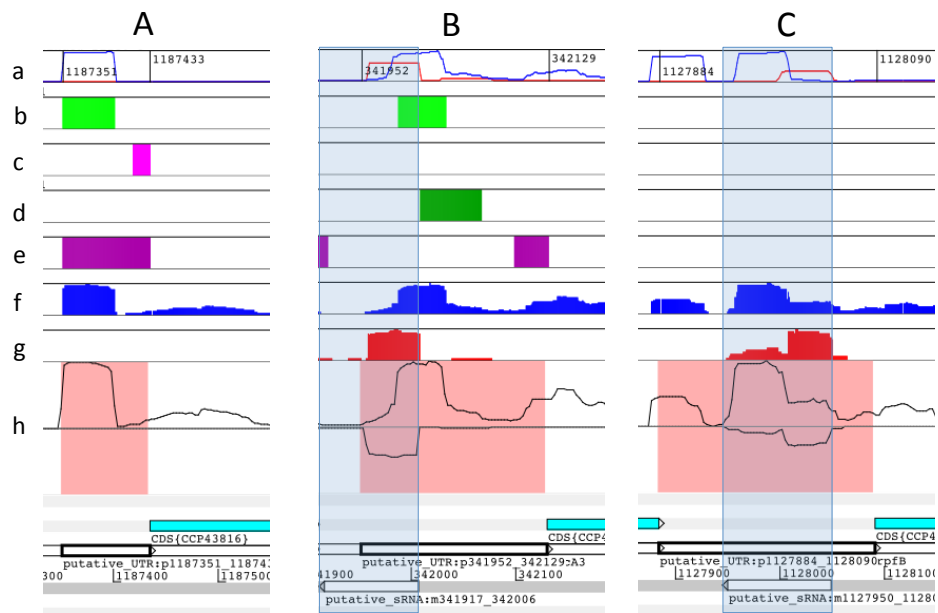
191 Our new method, *baerhunter*, allows the extraction of bacterial putative non-
192 coding expressed regions directly from RNA-seq data and facilitates the
193 integration of differential expression studies of coding and non-coding elements
194 in bacterial transcriptomes. Importantly, *baerhunter* compares favourably with
195 the most popular alternative method in tests with both simulated and real data.

196 References

- 197 Afgan,E. *et al.* (2018) The Galaxy platform for accessible, reproducible and
198 collaborative biomedical analyses: 2018 update. *Nucleic Acids Res*, **46**,
199 W537–W544.
- 200 Amman,F. *et al.* (2014) TSSAR: TSS annotation regime for dRNA-seq data. *BMC*
201 *Bioinformatics*, **15**, 89.
- 202 Arnvig,K.B. *et al.* (2011) Sequence-Based Analysis Uncovers an Abundance of
203 Non-Coding RNA in the Total Transcriptome of Mycobacterium
204 tuberculosis. *PLOS Pathogens*, **7**, e1002342.
- 205 Carver,T. *et al.* (2012) Artemis: an integrated platform for visualization and
206 analysis of high-throughput sequence-based experimental data.
207 *Bioinformatics*, **28**, 464–469.
- 208 Cortes,T. *et al.* (2013) Genome-wide Mapping of Transcriptional Start Sites
209 Defines an Extensive Leaderless Transcriptome in Mycobacterium
210 tuberculosis. *Cell Reports*, **5**, 1121–1131.
- 211 Frazee,A.C. *et al.* (2015) Polyester: simulating RNA-seq datasets with differential
212 transcript expression. *Bioinformatics*, **31**, 2778–2784.
- 213 Haning,K. *et al.* (2014) Small RNAs in mycobacteria: an unfolding story. *Front.*
214 *Cell. Infect. Microbiol.*, **4**.
- 215 Kalvari,I. *et al.* (2018) Rfam 13.0: shifting to a genome-centric resource for non-
216 coding RNA families. *Nucleic Acids Res*, **46**, D335–D342.
- 217 Love,M.I. *et al.* (2014) Moderated estimation of fold change and dispersion for
218 RNA-seq data with DESeq2. *Genome Biology*, **15**, 550.
- 219 McClure,R. *et al.* (2013) Computational analysis of bacterial RNA-Seq data.
220 *Nucleic Acids Res*, **41**, e140–e140.
- 221 Michaux,C. *et al.* (2014) Physiological roles of small RNA molecules. *Microbiology*,
222 **160**, 1007–1019.

- 223 Pellin,D. *et al.* (2012) A Genome-Wide Identification Analysis of Small Regulatory
224 RNAs in *Mycobacterium tuberculosis* by RNA-Seq and Conservation
225 Analysis. *PLOS ONE*, **7**, e32723.
- 226 Peña-Castillo,L. *et al.* (2015) Detection of bacterial small transcripts from rna-seq
227 data: a comparative assessment. In, *Biocomputing 2016*. WORLD
228 SCIENTIFIC, pp. 456–467.
- 229 Pfeifer-Sancar,K. *et al.* (2013) Comprehensive analysis of the *Corynebacterium*
230 *glutamicum* transcriptome using an improved RNAseq technique. *BMC*
231 *Genomics*, **14**, 888.
- 232 R Core Team (2018) R: A Language and Environment for Statistical Computing R
233 Foundation for Statistical Computing, Vienna, Austria.
- 234 Thorpe,H.A. *et al.* (2017) Comparative Analyses of Selection Operating on
235 Nontranslated Intergenic Regions of Diverse Bacterial Species. *Genetics*,
236 **206**, 363–376.
- 237 Toffano-Nioche,C. *et al.* (2013) Detection of non-coding RNA in bacteria and
238 archaea using the DETR'PROK Galaxy pipeline. *Methods*, **63**, 60–65.
- 239 Wang,Y. *et al.* (2015) An empirical strategy to detect bacterial transcript
240 structure from directional RNA-seq transcriptome data. *BMC Genomics*,
241 **16**, 359.
- 242 Wilms,I. *et al.* (2012) Deep sequencing uncovers numerous small RNAs on all
243 four replicons of the plant pathogen *Agrobacterium tumefaciens*. *RNA*
244 *Biology*, **9**, 446–457.
- 245
- 246

247 **Figure 1.**
 248 **Baerhunter's predictions of UTRs and small RNAs are supported by**
 249 **experimentally detected transcription start sites (TSS)**



250
 251 In all three figures (A, B and C), the panels arranged in rows are:
 252 (a) experimentally detected TSS (normalised counts per base from the Cortes *et al.* (Cortes *et al.*,
 253 2013)dataset);
 254 (b) and (c), sRNA and UTR predictions from Rockhopper run with default sensitivity, 0.5;
 255 (d) and (e), sRNA and UTR predictions from Rockhopper run with increased sensitivity, 0.2;
 256 (f) and (g) RNA-seq trace from Rockhopper mapping of reads for sample ERR262980 (blue:
 257 positive strand, red: negative strand);
 258 and (h) the RNA-seq trace corresponding to read coverage in sample ERR262980 mapped by our
 259 own pipeline (see supplemental Methods).
 260 Filled pink rectangles in row (h) highlight the putative UTR region as predicted by *baerhunter*.
 261 Transparent blue rectangles (panels B and C) highlight the two short RNAs predicted by
 262 *baerhunter* on the negative strand.

263
 264 **A.**
 265 The experimentally detected TSS (blue line in row a) supports a 90 nt 5' UTR for the
 266 uncharacterized protein Rv1065. The *baerhunter's* prediction is 84 nt long (region highlighted
 267 with pink in row h), closely following the RNA-seq trace. Rockhopper's prediction is similar when
 268 run with the more sensitive detection threshold of 0.2 (dark pink rectangle, row e) but run at the
 269 default 0.5, it splits the prediction to a "non-coding RNA" (green rectangle, row b) and a much
 270 shorter 5' UTR (bright pink, row c) that is not supported by TSS data.

271 **B.**
 272 The *baerhunter* program predicts a long (178 nt) 5' UTR ahead of the Rv0282 gene and an
 273 antisense RNA (90nt) partially overlapping this UTR. Both predictions are supported by
 274 experimentally detected TSS (blue and red lines on the positive and negative strand respectively;
 275 row a). Rockhopper, run with default precision parameters, predicts a non-coding RNA (light
 276 green; panel b) and no 5' UTR (row c), whereas when run at higher sensitivity, it predicts a non-
 277 coding RNA further downstream (dark green, row d) as well as a short 5' UTR (dark pink, row e),
 278 corresponding to a weaker TSS just ahead of the coding region of the gene (small blue peak, row
 279 a). In both cases, Rockhopper misses the antisense RNA that is clearly seen in the RNA-seq trace
 280 of the exponentially growing bacteria (black trace on the negative strand, row h and red-fill trace,
 281 row g).

282 **C.**
 283 The *baerhunter* program discovers both the 5'UTR ahead of the Rv1009 (*rpfB*) gene and the
 284 antisense RNA overlapping it on the negative strand (both of which have experimental TSS

285 support). Rockhopper does not predict any ncRNA features in the same region, although its own
286 mapping of reads results in clear expression on both strands (rows f and g).
287
288 The similarities between read coverage as reported by Rockhopper (rows f and g) and our own
289 pipeline (row h) indicate that differences between the Rockhopper and *baerhunter* predictions
290 are not due to differences in the way the reads were mapped to the reference genome but instead
291 are due to the different ways the programs identify expressed regions.