NxTrim: optimized trimming of Illumina mate pair reads.

Supplementary Materials

Jared O’Connell¹, Ole Schulz-Trieglaff¹, Emma Carlson², Matthew M. Hims², Niall A. Gormley² and Anthony J. Cox¹,†

¹Computational Biology Group, Illumina Cambridge Ltd., Chesterford Research Park, Little Chesterford, Essex CB10 1XL, United Kingdom
²Technology Development, Illumina Cambridge Ltd., Chesterford Research Park, Little Chesterford, Essex CB10 1XL, United Kingdom
†to whom correspondence should be addressed

August 6, 2014

1 Velvet command line parameters

We assembled the libraries generated by NxTrim using Velvet (version 1.2.10) with the following commands:

velveth output_dir ${k} -short -fastq.gz ${prefix}.se.fastq.gz
 -shortPaired2 -fastq.gz ${prefix}.pe.fastq.gz
 -shortPaired3 -fastq.gz ${prefix}.mp.fastq.gz
 -shortPaired4 -fastq.gz ${prefix}.unknown.fastq.gz

velvetg output_dir -exp_cov auto -cov_cutoff auto -shortMatePaired4 yes

where ${k} is the k-mer size used and ${prefix} is simply the sample name. We performed assemblies across a range of k-mers (21,119), choosing the assembly with the largest contig N50 for each sample. The -shortMatePaired4 yes argument flags the unknown library as possibly containing contaminants and was found to improve assembly quality.

For the MiSeq Reporter trimmed reads, we only have one library. We used similar commands:

velveth output_dir ${k} -shortPaired2 -fastq.gz ${prefix}.fastq.gz

velvetg output_dir -exp_cov auto -cov_cutoff auto -shortMatePaired2 yes

again we searched over the same range of k-mers, taking the largest contig N50 for each sample.

These commands were inspired by Dr. Torsten Seemann’s blog:

2 Adapter trimming logic

We now outline the logic behind our adapter trimming and virtual library creation routine. We describe the metric we use for adapter detection in section 2.1 and the library assignment of read pairs in section 2.2.

Figure S1: Enumeration of the different trimming scenarios. The blue areas represent genomic DNA whilst the yellow area is the artificial Nextera Mate Pair adapter sequence which needs to be removed. Genomic DNA on opposite sides of the adapter are from physically distant (~4kb on average) genomic locations with Reverse-Forward orientation. The arrows represent the sequence assayed by the reads R1 and R2. The dashed boxes represent the sequence that will be kept after trimming the adapter sequence from the reads. When present (and detected), the location of the adapter informs us about the orientation and physical distance of the reads allowing them to categorise them into virtual libraries. Cases A and B are the most common, resulting in an obvious mate-pair (MP) or paired-end (PE) read pair respectively. In case C no adapter was present, but the overlap between reads tells us that the reads are forward-reverse orientated with a paired end distance (and can optionally be joined into a longer read). Case D allows us to produce either an MP or PE pair, we choose the pair with the longest read lengths, the overhang is stored as a single ended read if it is long enough. In cases E and F we have no information about the read orientation, typically these are MP reads (E) but there is a small amount of PE contamination (F).
2.1 Adapter detection

We search for adapter sequence within a read pair \((R_1, R_2)\) containing \(L_R\) bases in each read. The Nextera Mate Pair adapter sequences are:

- \(A_1 = \text{CTGTCTCTTATACACATCT}\)
- \(A_2 = \text{AGATGTGATAAGAGACAG}\)

The concatenation of these two strings, \(A_1 + A_2\), is the yellow region in Figure S1 and it (or at least a substring of it) will be what is typically observed in one or both the reads in a pair. Due to substitution (and sometimes indel) errors in sequencing we need a detection routine that allows for imperfect matches. Note the adapter sequence is the reverse-complement of itself so we do not need to worry about strandedness. We describe the search for a single read \(R\) as the detection routine is the same for both reads.

For both \(A_1\) and \(A_2\), we slide a 19-mer window across \(R\) and compute the Hamming distance between each window and the adapter, accounting for partial matches to the adapter by allowing the window to shrink as small as 12-bases at the extremities of the read and comparing with the appropriate substring of the adapter. If the smallest distance is below a threshold derived from a similarity measure \(\rho = 0.85\), we consider the adapter detected and return the indices \((a, b)\) of \(R\) that contain it. Checking for each adapter separately may seem redundant, but we do this for two reasons:

- Occasionally, the DNA fragments manage to circularise with only one adapter present.
- Hamming distances will not allow us to detect adapter sequence containing indel errors. Since these are rare in Illumina data, the chances of seeing more than one across the 38bp merged adapter is extremely low. Checking each half separately allows the presence of one half (with an indel error) to be inferred by the presence of another (with no indel error).

Both of these outcomes are very rare, but the 19-mer adapter is long enough to provide high specificity so no accuracy is lost by only checking for one half of the adapter rather than the full 38bp length.

The algorithms are described formally below, we assume the Hamming distance function is already defined.
Algorithm 1 Adapter/overlap detection routines

function ALIGNSTRING(R, A)
    $MIN_d = L_A$
    $MIN_i = NA$
    for $i \in [-(L_A - L_M), L_R - (L_A - L_M)]$
        if $i < 0$
            $d = Hamming(R[0, i + L_A], A[L_A + i, L_A])$
            $L_C = i + L_A$
        else if $i > L_R$
            $d = Hamming(R[i, L_R], A[0, L_R - i])$
            $L_C = L_R - i$
        else
            $d = Hamming(R[i, i + L_A], A[0, L_A])$
            $L_C = L_A$
        end if
        if $d < ((1 - \rho) \times (L_C))$ and $d < MIN_d$ then
            $MIN_i = i$
            $MIN_d = d$
        end if
    end for
    return $MIN_i$
end function

function DETECTADAPTER(R)
    $a = ALIGNSTRING(R, A_1)$
    if $a \neq NA$ then
        return $(a, a + 2L_A)$
    end if
    $a = ALIGNSTRING(R, A_1)$
    if $a \neq NA$ then
        return $(a - L_A, a + L_A)$
    end if
    return $(NA, NA)$
end function

function OVERLAP($R_1, R_2$)
    for $i \in [0, L_R - L_M]$ do
        $d = Hamming(R_1[i, L_R], R_2[0, L_R - i])$
        if $d < ((1 - \rho) \times (L_R - i))$ then
            return True
        end if
    end for
    return False
end function
2.2 Virtual library classification

After detecting the adapter (if present) in each read we trim and categorise the read-pairs. We define $(a_1, b_1)$ and $(a_2, b_2)$ as the beginning $(a)$ and end $(b)$ indices of the adapter sequence for reads $R_1$ and $R_2$ respectively. Note that we allow $a$ to be negative and $b$ to exceed the read length ($b > L_R$). Since the adapter may only be partially sequenced at the start (or end) of a read. We also define the variable $L_M = 12$, reads smaller than $L_M$ after trimming will be discarded.

We first try to detect if the adapter is present in either (or both) reads. If it is no adapter is found, we check if the reads overlap. If the reads overlap the read-pair is PE (case C) else it is UNKNOWN (case E or F). If the adapter is found we return a MP if the majority of genomic DNA is on opposite ends of the adapter (case A), else if most of the genomic DNA occurs on the same side of the adapter we return a PE (case B). If the adapter splits a read near the centre, we may also return a separate single read in addition to the PE and SE library (case D).
A formal description of this logic follows:

**Algorithm 2** Adapter trimming and virtual library creation

```plaintext
function TrimReads(R₁,R₂)
    SE=NA
    PE=NA
    MP=NA
    UNKNOWN=NA
    (a₁,b₁) = findAdapter(R₁)
    (a₂,b₂) = findAdapter(R₂)
    if a₁ = NA and a₂ = NA then
        if Overlap(R₁,ReverseComplement(R₂)) then
            PE = (R₁,R₂)
            return (SE,PE,MP,UNKNOWN)
        end if
    end if
    if a₁ = NA and a₂ ≠ NA then
        UNKNOWN=(R₁,R₂)
    else if a₁ ≠ NA and a₂ < M then
        SE = R₁[0,a₁]
    else if a₁ = NA and a₂ < M then
        SE = R₂[0,a₂]
    else if a₁ < L_R and a₂ < L_R and a₂ = NA then
        MP = (R₁[0,a₁],R₂[0,a₂])
    else if a₁ < L_R and b₁ ≥ L_R and a₂ = NA then
        MP = (R₁[0,a₁],R₂)
    else if a₂ < L_R and b₂ ≥ L_R and a₁ = NA then
        MP = (R₁,R₂[0,a₂])
    else if b₁ < L_R and a₂ = NA then
        MP = ResolveOverhang(R₁,R₂,a₁,b₁)
    else if b₂ < L_R and a₁ = NA then
        MP = ResolveOverhang(R₂,R₁,a₂,b₂)
    end if
    MP = ReverseComplement(MP) ▷ Puts mate-pairs in Forward-Reverse orientation
    UNKNOWN = ReverseComplement(UNKNOWN)
    return (SE,PE,MP,UNKNOWN)
end function

function ResolveOverhang(R₁,R₂,a,b)
    SE=NA
    PE=NA
    MP=NA
    if a < (L_R - b) then ▷ PE has bigger reads than MP
        PE=(R₁[b,L_R],R₂)
    elseif a < M then
        SE=R₁[0,a] ▷ Create a SE if the overhang is big enough
    else
        MP=(R₁[0,a],R₂)
    end if
    if (L_r - b) ≥ M then ▷ MP has bigger reads than PE
        SE=R₁[b,L_R]
    elseif
        return (SE,PE,MP)
end function
```

6
### Abbreviation: Bcer
**Bacteria:** Bacillus cereus ATCC 10987  
**Accession ID:** NC_003909, NC_005707  

### Abbreviation: EcDH
**Bacteria:** Escherichia coli str. K-12 substr. DH10B  
**Accession ID:** NC_010473  

### Abbreviation: EcMG
**Bacteria:** Escherichia coli str. K-12 substr. MG1655  
**Accession ID:** NC_000913  

### Abbreviation: list
**Bacteria:** Listeria monocytogenes  
**Accession ID:** NC_003210  

### Abbreviation: meio
**Bacteria:** Meiothermus ruber DSM 1279  
**Accession ID:** NC_013946  

### Abbreviation: ped
**Bacteria:** Pedobacter heparinus DSM 2366  
**Accession ID:** NC_013061  

### Abbreviation: pneu
**Bacteria:** Klebsiella pneumoniae subsp. pneumoniae MGH 78578  
**Accession ID:** NC_009648, NC_009649, NC_009650, NC_009651, NC_009652, NC_009653  

### Abbreviation: rhod
**Bacteria:** Rhodobacter sphaeroides 2.4.1  
**Accession ID:** NC_007488, NC_007489, NC_007490, NC_007493, NC_007494, NC_009007, NC_009008  
**NCBI FTP:** [ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Rhodobacter_sphaeroides_2_4_1_uid57653/](ftp.ncbi.nlm.nih.gov/genomes/Bacteria/Rhodobacter_sphaeroides_2_4_1_uid57653/)

### Abbreviation: TB
**Bacteria:** Mycobacterium tuberculosis H37Ra  
**Accession ID:** NC_009525  

### Table S1: Summary of bacteria analysed and the relevant NCBI information on their reference genomes.

There were two repeats of each strain. All 18 samples were prepared with the Nextera Mate Pair protocol and sequenced in a single MiSeq run using 2x151bp reads. The untrimmed reads we used as input to NxTrim (3.9Gbp in all) are available from BaseSpace via [https://basespace.illumina.com/s/TXv32Ve6wTl9](https://basespace.illumina.com/s/TXv32Ve6wTl9) (free registration required).
<table>
<thead>
<tr>
<th>Reference</th>
<th>Coverage</th>
<th>k-mer</th>
<th>Reference</th>
<th>Contig</th>
<th>Contig</th>
<th>#contigs</th>
<th>Scaffold</th>
<th>Scaffold</th>
<th>#scaffolds</th>
<th>Assembly</th>
<th>% Full genes assembled</th>
</tr>
</thead>
<tbody>
<tr>
<td>size (kb)</td>
<td>depth</td>
<td>size</td>
<td>recovered</td>
<td>NG50 (kb)</td>
<td>NG50 (kb)</td>
<td></td>
<td>NG50 (kb)</td>
<td>NG50 (kb)</td>
<td></td>
<td>length (kb)</td>
<td></td>
</tr>
<tr>
<td>Bcer1</td>
<td>5432.65</td>
<td>23.99</td>
<td>43</td>
<td>98.76</td>
<td>88.27</td>
<td>88.27</td>
<td>125</td>
<td>3869.19</td>
<td>11523.36</td>
<td>22</td>
<td>9515.68</td>
</tr>
<tr>
<td>Bcer2</td>
<td>5432.65</td>
<td>27.16</td>
<td>55</td>
<td>98.80</td>
<td>107.78</td>
<td>107.78</td>
<td>100</td>
<td>1656.36</td>
<td>650.25</td>
<td>20</td>
<td>5384.93</td>
</tr>
<tr>
<td>EcDH1</td>
<td>4866.14</td>
<td>48.19</td>
<td>65</td>
<td>96.97</td>
<td>224.96</td>
<td>224.96</td>
<td>59</td>
<td>4185.82</td>
<td>744.63</td>
<td>21</td>
<td>4553.46</td>
</tr>
<tr>
<td>EcDH2</td>
<td>4866.14</td>
<td>31.12</td>
<td>59</td>
<td>96.60</td>
<td>165.50</td>
<td>165.50</td>
<td>78</td>
<td>2351.56</td>
<td>615.09</td>
<td>27</td>
<td>4538.10</td>
</tr>
<tr>
<td>EcMG1</td>
<td>4641.65</td>
<td>35.44</td>
<td>61</td>
<td>99.10</td>
<td>284.59</td>
<td>284.59</td>
<td>66</td>
<td>4594.81</td>
<td>442.95</td>
<td>21</td>
<td>4605.78</td>
</tr>
<tr>
<td>EcMG2</td>
<td>4641.65</td>
<td>35.59</td>
<td>58</td>
<td>98.96</td>
<td>202.96</td>
<td>202.96</td>
<td>72</td>
<td>4598.59</td>
<td>420.34</td>
<td>19</td>
<td>4604.36</td>
</tr>
<tr>
<td>list1</td>
<td>2944.53</td>
<td>67.90</td>
<td>97</td>
<td>99.69</td>
<td>1055.30</td>
<td>1572.90</td>
<td>9</td>
<td>2927.87</td>
<td>2189.51</td>
<td>4</td>
<td>2933.10</td>
</tr>
<tr>
<td>list2</td>
<td>2944.53</td>
<td>53.34</td>
<td>81</td>
<td>99.59</td>
<td>2424.01</td>
<td>2424.01</td>
<td>6</td>
<td>2934.64</td>
<td>4242.04</td>
<td>5</td>
<td>2934.80</td>
</tr>
<tr>
<td>meio1</td>
<td>3097.46</td>
<td>54.92</td>
<td>73</td>
<td>99.89</td>
<td>300.90</td>
<td>300.90</td>
<td>30</td>
<td>3000.26</td>
<td>1271.26</td>
<td>16</td>
<td>3103.92</td>
</tr>
<tr>
<td>meio2</td>
<td>3097.46</td>
<td>47.80</td>
<td>75</td>
<td>99.89</td>
<td>192.67</td>
<td>192.67</td>
<td>37</td>
<td>3002.14</td>
<td>1283.90</td>
<td>13</td>
<td>3101.21</td>
</tr>
<tr>
<td>ped1</td>
<td>5167.38</td>
<td>35.95</td>
<td>49</td>
<td>99.65</td>
<td>459.30</td>
<td>456.76</td>
<td>48</td>
<td>5147.15</td>
<td>1271.26</td>
<td>16</td>
<td>5163.98</td>
</tr>
<tr>
<td>ped2</td>
<td>5167.38</td>
<td>27.29</td>
<td>58</td>
<td>99.62</td>
<td>290.96</td>
<td>290.96</td>
<td>62</td>
<td>5154.64</td>
<td>3209.92</td>
<td>13</td>
<td>5164.31</td>
</tr>
<tr>
<td>pneu1</td>
<td>5694.89</td>
<td>32.54</td>
<td>57</td>
<td>97.19</td>
<td>171.75</td>
<td>169.84</td>
<td>129</td>
<td>5290.29</td>
<td>546.70</td>
<td>33</td>
<td>5547.76</td>
</tr>
<tr>
<td>pneu2</td>
<td>5694.89</td>
<td>29.31</td>
<td>65</td>
<td>97.47</td>
<td>170.26</td>
<td>170.26</td>
<td>148</td>
<td>3624.50</td>
<td>499.86</td>
<td>44</td>
<td>5566.88</td>
</tr>
<tr>
<td>rhod1</td>
<td>4602.98</td>
<td>39.87</td>
<td>61</td>
<td>97.68</td>
<td>177.75</td>
<td>177.75</td>
<td>90</td>
<td>4126.89</td>
<td>2503.60</td>
<td>14</td>
<td>4499.77</td>
</tr>
<tr>
<td>rhod2</td>
<td>4602.98</td>
<td>45.09</td>
<td>69</td>
<td>97.89</td>
<td>272.18</td>
<td>272.18</td>
<td>74</td>
<td>3189.60</td>
<td>2503.60</td>
<td>14</td>
<td>4513.78</td>
</tr>
<tr>
<td>TB1</td>
<td>4419.98</td>
<td>46.86</td>
<td>69</td>
<td>97.83</td>
<td>98.39</td>
<td>75.44</td>
<td>100</td>
<td>4595.05</td>
<td>154.71</td>
<td>23</td>
<td>4539.53</td>
</tr>
<tr>
<td>TB2</td>
<td>4419.98</td>
<td>38.60</td>
<td>53</td>
<td>98.03</td>
<td>116.65</td>
<td>116.65</td>
<td>97</td>
<td>4361.85</td>
<td>154.83</td>
<td>23</td>
<td>4367.03</td>
</tr>
<tr>
<td>Average</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table S2: Assembly metrics for Velvet assemblies when using reads trimmed with NxTrim.
Table S4: Breakdown of the proportions of different virtual libraries generated by our trimming method. Note the sum of MP, UNKNOWN and PE constitute 100% of the read pairs that passed standard chastity/purity filters. MP+UNKNOWN is the proportion of reads that will have large mate-pair insert sizes (unknown pairs have a small amount of paired-end contamination). SE is percentage of pairs where a third “unpaired” single read was generated from an overhang of >21bp. Total is the average across all samples weighted be coverage. So typically we see ≈62.57% of reads with mate-pair orientation (with some contamination), ≈37.43% of reads with paired-end orientation and ≈18.89% of either of these pairs generate an orphaned overhanged single read.

<table>
<thead>
<tr>
<th>Sample</th>
<th>MP %</th>
<th>UNKNOWN %</th>
<th>MP+UNKNOWN%</th>
<th>PE%</th>
<th>SE%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bcer1</td>
<td>53.67</td>
<td>10.06</td>
<td>63.73</td>
<td>36.27</td>
<td>42.67</td>
</tr>
<tr>
<td>Bcer2</td>
<td>49.13</td>
<td>13.71</td>
<td>62.84</td>
<td>37.16</td>
<td>40.08</td>
</tr>
<tr>
<td>EcDH1</td>
<td>43.95</td>
<td>19.48</td>
<td>63.43</td>
<td>36.57</td>
<td>37.65</td>
</tr>
<tr>
<td>EcDH2</td>
<td>45.58</td>
<td>17.46</td>
<td>63.03</td>
<td>36.97</td>
<td>38.02</td>
</tr>
<tr>
<td>EcMG1</td>
<td>45.68</td>
<td>17.97</td>
<td>63.65</td>
<td>36.35</td>
<td>38.71</td>
</tr>
<tr>
<td>EcMG2</td>
<td>40.89</td>
<td>25.06</td>
<td>65.96</td>
<td>34.04</td>
<td>35.43</td>
</tr>
<tr>
<td>list1</td>
<td>41.34</td>
<td>23.78</td>
<td>65.12</td>
<td>34.88</td>
<td>36.52</td>
</tr>
<tr>
<td>list2</td>
<td>46.21</td>
<td>19.11</td>
<td>65.32</td>
<td>34.68</td>
<td>38.72</td>
</tr>
<tr>
<td>meio1</td>
<td>49.08</td>
<td>13.44</td>
<td>62.52</td>
<td>37.48</td>
<td>40.14</td>
</tr>
<tr>
<td>meio2</td>
<td>40.91</td>
<td>24.61</td>
<td>65.51</td>
<td>34.49</td>
<td>35.24</td>
</tr>
<tr>
<td>ped1</td>
<td>45.00</td>
<td>18.98</td>
<td>63.98</td>
<td>36.02</td>
<td>38.50</td>
</tr>
<tr>
<td>ped2</td>
<td>48.99</td>
<td>15.01</td>
<td>64.00</td>
<td>36.00</td>
<td>39.95</td>
</tr>
<tr>
<td>pneu1</td>
<td>38.05</td>
<td>28.00</td>
<td>66.05</td>
<td>33.95</td>
<td>33.62</td>
</tr>
<tr>
<td>pneu2</td>
<td>40.43</td>
<td>24.55</td>
<td>64.98</td>
<td>35.02</td>
<td>35.39</td>
</tr>
<tr>
<td>rhod1</td>
<td>43.50</td>
<td>18.45</td>
<td>61.94</td>
<td>38.06</td>
<td>37.94</td>
</tr>
<tr>
<td>rhod2</td>
<td>40.71</td>
<td>22.10</td>
<td>62.81</td>
<td>37.19</td>
<td>35.21</td>
</tr>
<tr>
<td>TB1</td>
<td>43.45</td>
<td>16.90</td>
<td>60.35</td>
<td>39.65</td>
<td>36.78</td>
</tr>
<tr>
<td>TB2</td>
<td>43.77</td>
<td>18.16</td>
<td>61.93</td>
<td>38.07</td>
<td>37.53</td>
</tr>
<tr>
<td>Average</td>
<td>44.24</td>
<td>19.42</td>
<td>63.66</td>
<td>36.34</td>
<td>37.51</td>
</tr>
</tbody>
</table>

Table S4: Breakdown of the proportions of different virtual libraries generated by our trimming method. Note the sum of MP, UNKNOWN and PE constitute 100% of the read pairs that passed standard chastity/purity filters. MP+UNKNOWN is the proportion of reads that will have large mate-pair insert sizes (unknown pairs have a small amount of paired-end contamination). SE is percentage of pairs where a third “unpaired” single read was generated from an overhang of >21bp. Total is the average across all samples weighted be coverage. So typically we see ≈62.57% of reads with mate-pair orientation (with some contamination), ≈37.43% of reads with paired-end orientation and ≈18.89% of either of these pairs generate an orphaned overhanged single read.
Figure S2: Plot of the number of genes against contig N50 (log$_{10}$ scale in kb) for different assemblies across all samples. Assemblies were performed for all odd k-mers between 21 and 121 which generated assemblies with varying contig N50s. We chose the assemblies with the highest contig N50 which appears to be a reasonable criteria in this scenario, given that contig N50 is strongly correlated with the number of genes found. Scaffold N50 was found to be a less reliable metric in this setting.
Figure S3: Insert sizes of each of the virtual libraries.
Figure S4: Alignments of each scaffold against the respective reference genome for the NxTrim assemblies.

12