

For pairing there are 3 steps (run each of them in screen mode):

1 - radsort both match files

```
radsort F3.all_chromosomes.csfasta.ma.25.2 6 >
F.all_chromosomes.csfasta.ma.25.2.sorted
radsort R3.all_chromosomes.csfasta.ma.25.2 6 >
R.all_chromosomes.csfasta.ma.25.2.sorted
```

2 - split up the sorted match files

```
split_sorted_match_files_into_groups_of_panels.pl
```

```
-f3 <f3_match_file_sorted>
-r3 <r3_match_file_sorted>
-n <number_of_panels_per_group>
-dir <output_directory>
```

-f3 and -r3 are the sorted match files that you made in step 1
-n we normally use 10
-dir is where you want the output to go - such as pairing.ml

3 - generate running scripts

Usage: /share/apps/corona_lite/bin/pairing_by_panel_save_script.pl

REQUIRED

```
-dir_f3          <f3_panel_directory>
-dir_r3          <r3_panel_directory>
-dir_scripts     <saved_scripts_directory : default scripts>
-insert_start    <insert_start>
-insert_end      <insert_end>
-ref             <reference_sequence>
-e              <total_number_of_errors>
-dir            <output_directory>
```

OPTIONAL

```
-mask_f3        <f3_mask>
-mask_r3        <r3_mask>
-errors         <pairing_to_mates_errors>
-format         <reference sequence: single or multiple :
```

default single,

should be set to multiple if running against

multiple chromosomes>

```
-panel_start     <panel_start: default is 1>
-panel_end       <panel_end: default is 2500>
-unique_only     <yes or no: default is yes>
-mismatch_threshold
```

```
-dir_f3 is the F3_match_files directory created in step 2
-dir_r3 is the R3_match_files directory created in step 2
-insert_start and -insert_end are the pairing range
-e (I usually add the number of mismatches allowed in each
```

tag)

```
-dir wherever you want the output to go, I usually do
pairing_500_800 or something in pairing.ml
```

```
-ref for multi-chromosome references give it a multi-entry
fasta file in the same order that was
used in the cmap file for matching. If you don't give it a
reference sequence then pairing will
be done without mate pair rescue
-format multiple (as long as the reference sequence is in
```

multi-entry format)

You can use `-panel_start` and `-panel_end` to limit the number of panels which is nice if you just want to sample the data to find the appropriate insert size. It's actually groups of panels that it uses so if you used `-n 10` in step 2 and use `-panel_start 1` and `-panel_end 10` here then it will use 100 panels

The other options shouldn't be needed.

Note: The reference genome file should be a multi-fasta file with sequences in the same order specified by the `chr_ID` (1st) column in the `cmap` file. If you are not sure about the concatenated reference file, you can use the script `concatenate_chr_seqs_cmap.pl` to make it:

```
concatenate_chr_seqs_cmap.pl
```

Concatenate chr seqs for pairing using `cmap` file.

Usage:

```
./concatenate_chr_seqs_cmap.pl [-h] -c cmap_file [-o outfile]
```

```
-h          : Print usage
-c cmap    : cmap file
-o file    : Output file name, default to STDOUT
```

4. The saved scripts from step 3 can be run following instructions described in `RELEASE_NOTES`.

5. Pairing stats:

There is a simple script `pairing_counts.pl` that will give you the total mates, total good mates, unique mates and unique good mates numbers.

To run the script, just `cd` into the pairing run dir where `F3_R3.mates` and `F3_R3.mates.unique` files are located and type the command:

```
pairing_counts.pl
```

The numbers will be output to `STDOUT` as well as in a file called `counts.dat`.