细胞器基因组重组检测与重组体图谱绘制教程

——软件MiRI与MiRIV的使用

# 一、软件安装

## 1、下载软件压缩包

网址为：https://github.com/wlqg1983/MiRI\_MiRIV\_1.0

## 2、解压压缩包并进入软件文件夹内：

unzip MiRI\_MiRIV\_1.0-main.zip

cd MiRI\_MiRIV\_1.0-main

## 3、安装并激活 MiRI 与 MiRIV 运行所需的 conda 环境：

conda env create -f MiRI\_MiRIV\_1.0.yml

conda activate MiRI\_MiRIV\_1.0

## 4、更新部分程序

sh install.sh

rm install.sh

## 5、验证安装结果

python bin/MiRI.py -h

**usage:** MiRI.py [-h] -c CONFIG [-redo] [-resume] [-v]

**MiRI:** A tool to check spanning reads for supporting subconfig of your organelle genome.

**options:**

-h, --help show this help message and exit

-c, CONFIG Path to external configuration file.

-redo Delete all previous results and start calculation anew.

-resume Resume from a previous project.

-v, --version Show the version number and exit.

python bin/MiRIV.py -h

**usage:** MiRIV.py [-h] -c CONFIG [-redo] [-v]

**MiRIV:** A tool to map the confgiure of your organelle genome.

**options:**

-h, --help show this help message and exit

-c, CONFIG Path to external configuration file.

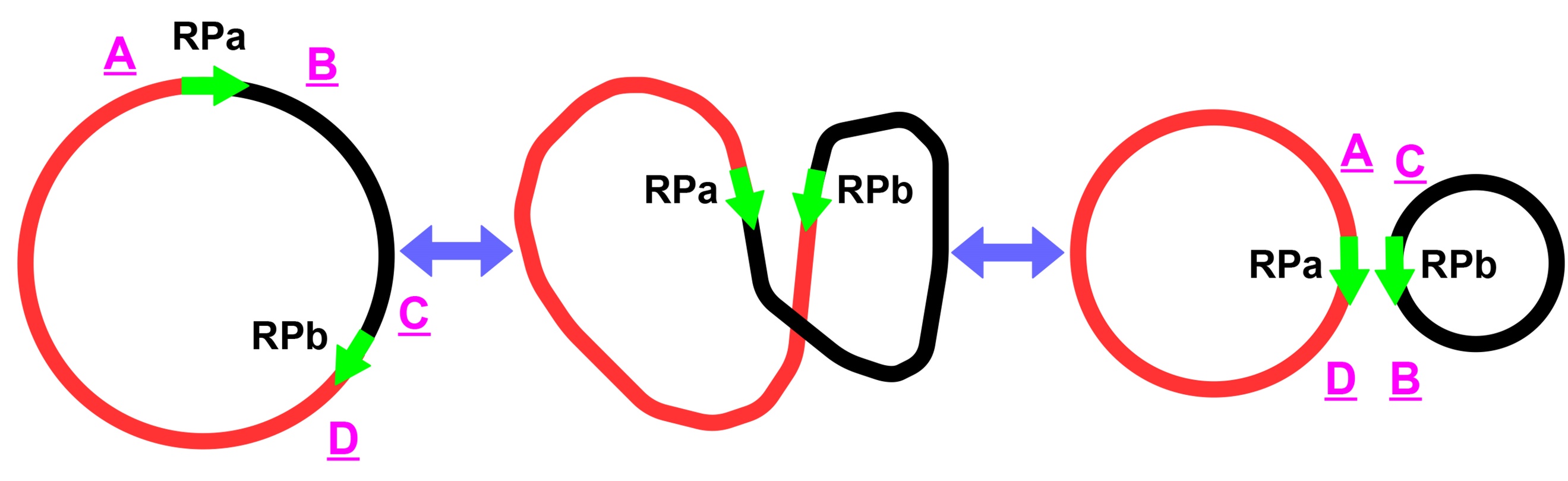
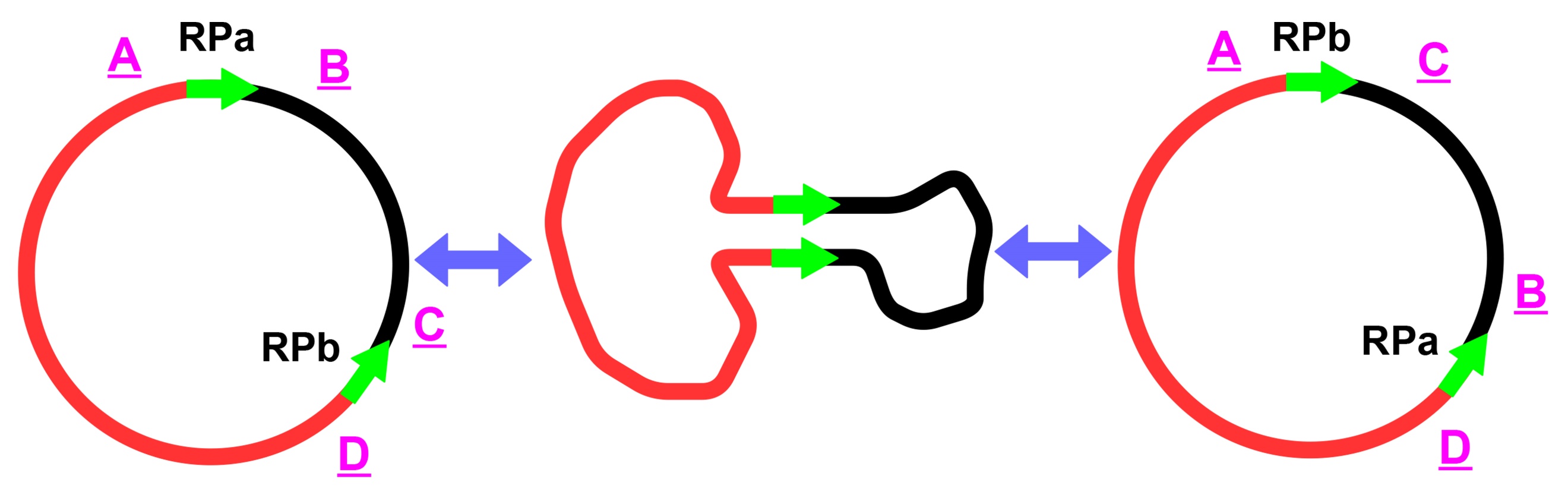
-redo Delete all previous results and start calculation anew.

-v, --version Show the version number and exit.

# 二、软件的运行原理

## 1、MiRI 的运行原理

MiRI（**Mi**togenome **R**ecombination **I**dentification）设计的运行原理是反向重复序列（IRs）的重组会使中间序列发生倒位（图 2-1A），而涉及正向重复序列（DRs）的重组则会产生一对亚基因组分子（图 2-1B）。



**B**

**A**

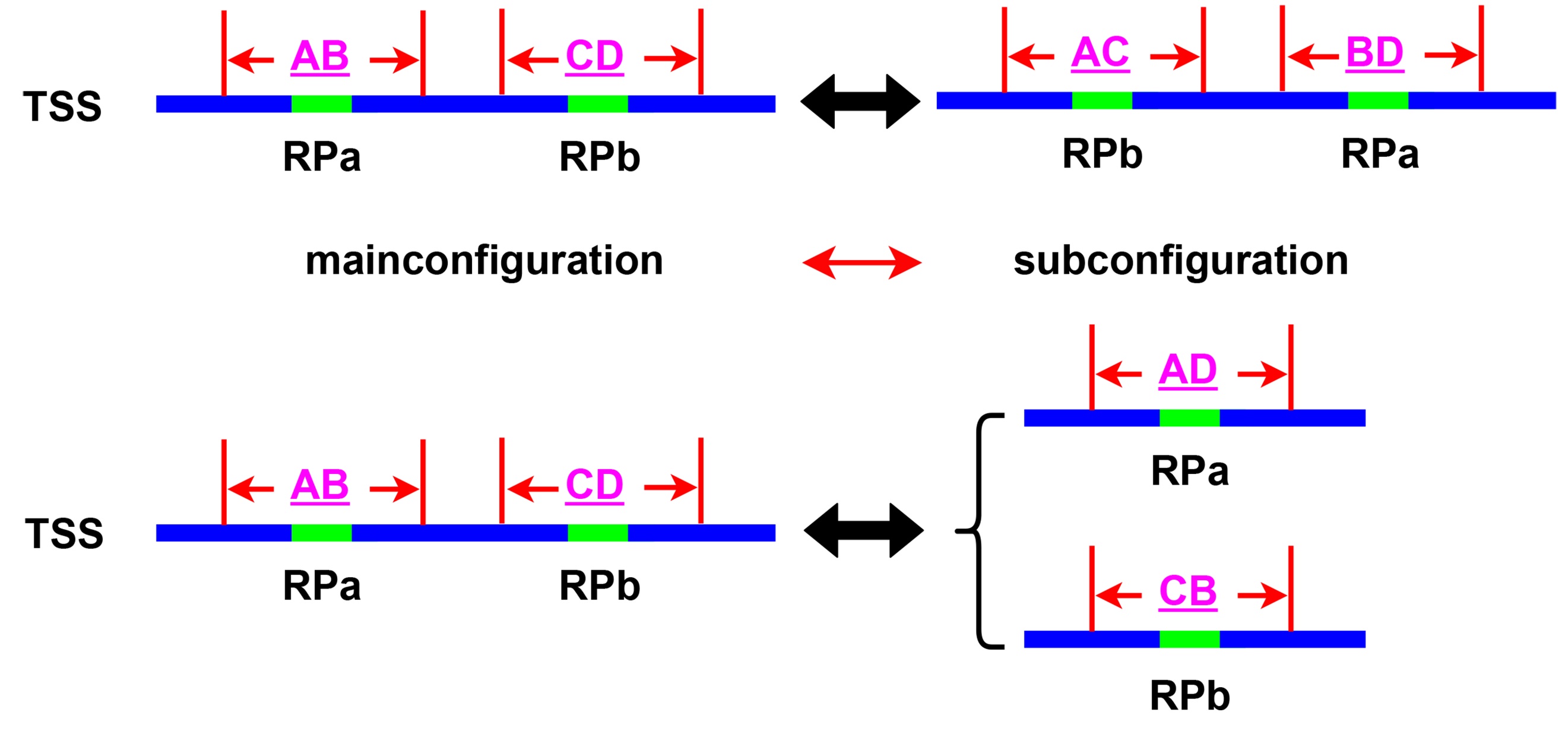
**B**

**A**

**图 2-1. 重复序列介导的环状基因组重组示意图。**

A：由反向重复序列（IRs）介导的重组，B：由正向重复序列（DRs）介导的重组。

以重复序列为中心，分别从主要构型和次要构型中截取一段序列（trimmed a short sequence, TSS），如图2-2所示，主要构型中分别以成对的重复序列RPa与RPb为中心，截取TSS为AB与CD，标记为TSSAB与TSSCD。IR介导的次要构型中，截取TSSAC与TSSBD，分别以重复序列RPb和RPa为中心（图1-1A）。DR介导的次要构型中，截取TSSAD与TSSCB，分别以重复序列RPa和RPb为中心（图2-2B）。



**A**

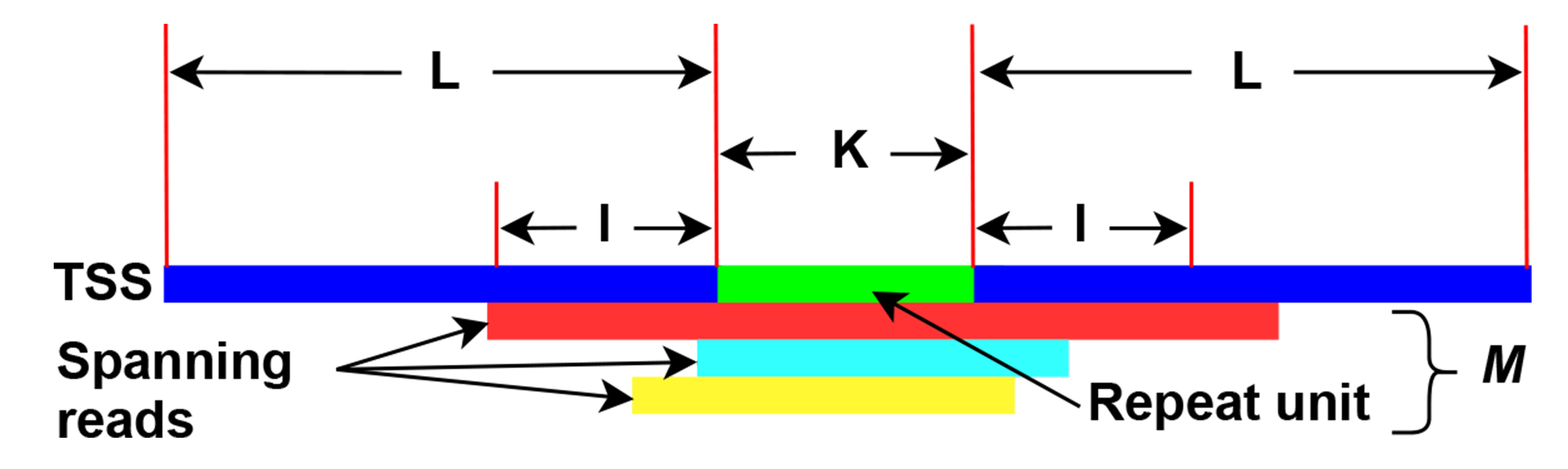
**A**

**B**

**图 2-2 转录起始位点（TSS）截取示意图**

涉及反向重复序列（IRs）的重组会使中间序列发生倒位（A），而涉及正向重复序列（DRs）的重组则会产生一对亚基因组分子（B）。AB、CD、AD、CB、AC 和 BD 代表转录起始位点（TSS）。AB 和 CD 来自主构型。AD 和 CB 来自由正向重复序列（DRs）介导的亚构型。AC 和 BD 来自由反向重复序列（IRs）介导的亚构型。

在主要构型和次要构型中，以重复序列为中心，左右各截取*L*个碱基，获得TSS后，MiRI将测序的read映射至TSS（图2-3），然后从映射至TSS的read中查找可以跨越重复序列的read（*l* > 0），若有read跨越了重复序列（即*M* > 0），则认为该TSS对应的基因组构型存在。



**图 2-3 读取序列映射到转录起始位点（TSS）的示意图**（trimmed a short sequence）。

*L*：表示在一个重复单元两侧截取的序列长度。默认值为 1000 碱基对（bp）。

*K*：一个重复单元的长度。默认值为 50 碱基对（bp）。

*I*：表示一条读取序列跨越一个重复单元左右两侧的长度。当 *I* ≥ 1 碱基对（bp）时，该读取序列被认为跨越了一个重复单元。

*M*：跨越一个重复单元且 *I* ≥ 1 碱基对（bp）的读取序列的数量。

## 2、重组率的定义与计算

在计算重复序列介导的线粒体基因组重组的概率时，主要构型中，跨越 TSSAB 和 TSSCD 中的 RPa 与 RPb 重复序列的read数量记录为 *M*AB 与 *M*CD。次要构型中，跨越 TSSAD 和 TSSCB，以及 TSSAC 和 TSSBD 中的 RPa 与 RPb 重复序列的read数量记录为 *M*AD 与 *M*CB，以及 *M*AC 与 *M*BD。

DR介导基因组重组中，RPa介导重组的概率为：



以上是考虑单链（2-1）和双链（2-2）两种情况下重组率的计算公式。

DR介导基因组重组中，RPb介导重组的概率为：



以上是考虑单链（2-3）和双链（2-4）两种情况下重组率的计算公式。

IR介导基因组重组中，RPa介导重组的概率为：



以上是考虑单链（2-5）和双链（2-6）两种情况下重组率的计算公式。

IR介导基因组重组中，RPb介导重组的概率为：



以上是考虑单链（2-7）和双链（2-8）两种情况下重组率的计算公式。

# 三、MiRI 软件的运行

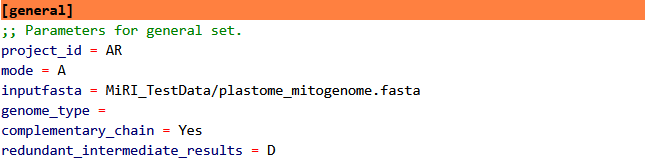
python bin/MiRI.py -c MiRI.config.ini

MiRI运行需要较多的参数，配置文件 MiRI.config.ini 用于设置各种参数，但多数参数可以采用默认值，仅少数几个参数需要设置。

## 1、MiRI 的运行模式一

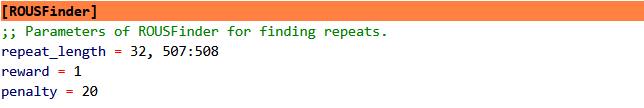
当 mode = A 时（图 3-1），为 MiRI 的第一种运行模式。MiRI 将先从基因组内查找正向和反向重复序列，然后再检测可以介导基因组重组的重复序列对。此时，用户必须提供基因组序列文件，查找的重复序列的长度，测序文件以及认定次要构型存在的参数。

基因组序列文件（inputfasta）为 fasta 格式，同时需要指明基因组的类型（genome\_type）为线性（L）还是环状（C）的（图 3-1）。当基因组含多条染色体时，将所有的染色体置于同一个 fasta 文件内。



**图 3-1 [general] 中参数的设置**

查找的重复序列的长度（repeat\_length），即图2-2中的K值，可设置为一个区间，如50 bp ≤ 长度 ≤ 1000 bp，可设置为50:1000；长度 ≥ 50 bp，可设置为 50: ；也可以设置几个长度，如50bp、100bp，可设置为50, 100（图 3-2）。



**图 3-2 [ROUSFinder] 中重复序列长度的设置**

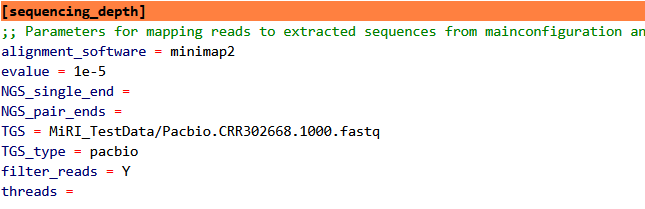
对于测序数据，MiRI 可接受二代（NGS）和三代（TGS）数据，NGS 和 TGS 的数据一次仅能接受其中一种（图 3-3）。提供双端数据时，双端数据的文件要以空格隔开。当提供TGS数据时，还要指明数据来自Nanopore测序平台（ont）还是Pacbio测序平台（pacbio）（图 3-3）。

检测次要构型存在的参数主要是spanning\_read\_flanking\_repeat\_length 和spanning\_read\_number （图 3-4）。这是判定次要构型是否存在的重要参数，默认值均为1，后续的重过滤模式（refilter\_mode = Y）中可对其重新设置，以对查询的重复序列介导基因组的重组结果进行多次筛选。

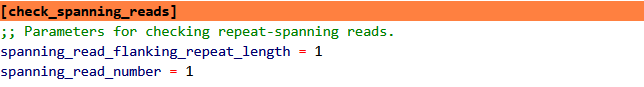
比对软件中，minimap2、bwa和blast的适用场景如表 3-1 所示。

**表 3-1 minimap2、bwa和blast适用场景比较**

|  |  |  |
| --- | --- | --- |
| **工具** | **最佳应用场景** | **可能漏检原因** |
| minimap2 | 长读长、基因组组装、快速比对 | 短读长、高重复区域、默认参数宽松 |
| BWA | 短读长、变异检测、精确比对 | 长读长处理效率低 |
| BLAST | 同源性搜索、跨物种比较 | 计算耗时，不适合大规模比对 |



**图 3-3 [ROUSFinder] 中重复序列长度的设置**

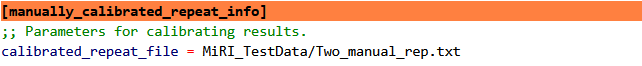


**图 3-4 [check\_spanning\_reads] 中次要构型存在的认定参数**

spanning\_read\_flanking\_repeat\_length是read跨越重复序列后的长度，即图 2-3 中的*l*值。spanning\_read\_number是跨越重复序列后的长度≥l的read的数量，即图 2-3 中的 *M* 值。

## 2、MiRI 的运行模式二

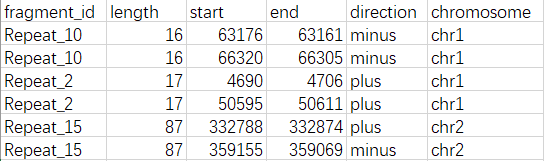
查询重复序列的工作极具挑战，不同的重复序列结果，对介导基因组重组的结果影响较大，而且不同算法间查找的重复序列的结果不同。所以，MiRI设置了可接受用户提供重复序列信息的接口（图 3-5），允许用户自己提供重复序列。此时，需设置mode = C（图 3-6）。用户提供的重复序列信息文件为tsv格式（图 3-7）。相同的重复序列，采用相同的fragment\_id。当基因组仅有一条染色体的时候，需要去掉chromosome列。



**图 3-5 [manually\_calibrated\_repeat\_info] 中用户提供重复序列信息**



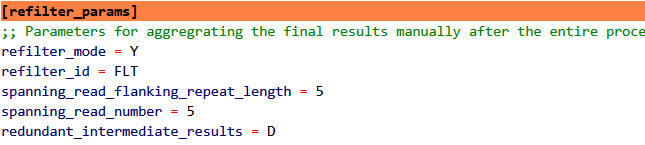
**图 3-6 MiRI 进入模式二的设置参数**



**图 3-7** **用户提供的重复序列信息示例（tsv格式）**

## 3、MiRI 的运行模式三

若对初次设置的筛选条件不满意，用户可设置refilter\_mode = Y （图 3-8），对 spanning\_read\_flanking\_repeat\_length 和 spanning\_read\_number 重新进行设置，对查询结果重新进行清洗，以获取更满意的结果。

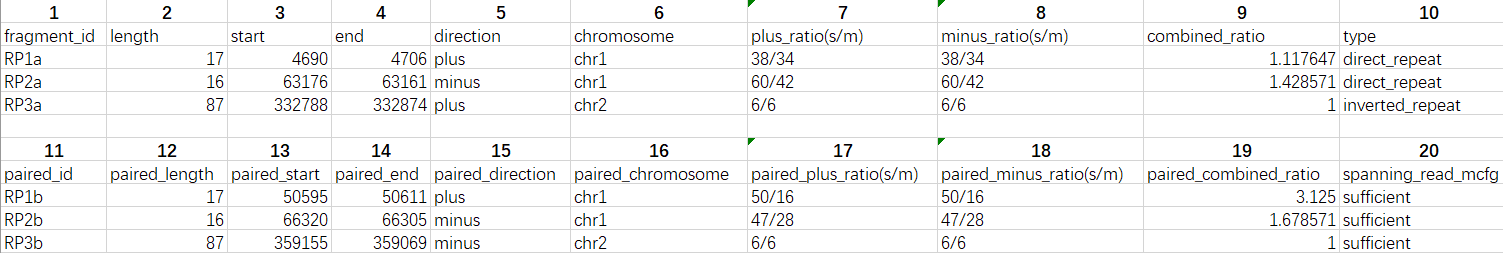


**图 3-8 [refilter\_params] 中过滤条件的重新设置**

# 四、MiRI 核心结果解读

MiRI的运行结果存储在文件夹{project\_id}/final\_repeat-spanning\_results\_{project\_id}内的 paired\_repeats\_recomb-supporting\_ratio.tsv 文件内。

该结果为一个20列的tsv文件，如图 4-1 所示。



**图 4-1 介导基因组重组的重复序列的信息**

每一列的含义如下：

① fragment\_id: 重复单元的编号。

② length: 重复单元的长度。

③ start: 重复单元在基因组中的起始位置。

④ end: 重复单元在基因组中的终止位置。

⑤ direction: 在 DNA 正链或负链上的位置。

⑥ chromosome: 基因组中某条染色体的编号。

⑦ plus\_ratio(s/m): 在 DNA 正链上，亚构型（s）和主构型（m）中跨越重复序列的读取序列数量之比。

⑧ minus\_ratio(s/m): 在 DNA 负链上，亚构型（s）和主构型（m）中跨越重复序列的读取序列数量之比。

⑨ combined\_ratio: DNA 两条链上重复序列介导的基因组重组的总体比例。

⑩ type: 重复序列的类型（正向重复或反向重复）。

⑪ spanning\_read\_mcfg: 主要构型中，跨越重复序列的read数量是否符合用户设置的数量。

注：“配对项” 指的是介导基因组重组的一对重复单元中的另一个重复单元。

# 五、MiRI 配置文件的详解

用户可通过更加详细的设置 .ini 配置文件来挖掘MiRI的性能，表 5-1 是对 .ini配置文件各参数的详细解读。

**表 5-1 配置文件中参数详解**

|  |  |  |
| --- | --- | --- |
| 参数类别 | 参数 | 取值与含义 |
| [general] | project\_id (必选参数) | 项目编号，由字母、数字和下划线组成 |
| mode (默认值 A) | 软件运行的模式，取值为\*N/A/R/C\* (大小不敏感)。N：程序不运行；A：程序自动运行；R：仅运行ROUSFinder软件查找重复序列；C：从用户提供的重复序列中查找跨重复序列的read，此时[calibrate\_ROUSFinder\_results]类别中的“calibrated\_repeat\_file”必须提供。 |
| inputfasta (必选参数) | 细胞器基因组序列文件，可包含多条染色体 |
| genome\_type (默认值 C) | 设置基因组为线性 (L) 或环状 (C) |
| complementary\_chain (默认值 Y) | 查找跨越重复序列的 read 时，考虑DNA 的双链 (Y) 或不考虑 (N) |
| redundant\_intermediate\_results (默认值 D) | 软件运行时，删除中间结果 (D) 或者不删除 (K) |
| [ROUSFinder] | repeat\_length (默认值 50: ) | 重复序列长度区间。长度 ≥ 50bp，设置为 50:。长度 ≤ 100bp，设置为 :100。100bp ≤ 长度 ≤ 200bp，设置为 100:200。最小值为 5bp。 |
| reward (默认值 1) | ROUSFinder参数，查找重复序列时，序列比对的奖励值 |
| penalty (默认值 20) | ROUSFinder参数，查找重复序列时，序列比对的惩罚值 |
| [manually\_calibrated\_repeat\_info] | calibrated\_repeat\_file | 手工校准重复序列结果的输入文件位置，mode = C 时为必选参数 |
| [mainconfiguration] | flanked\_sequence\_length (默认值 1000bp) | 在主要构型中，重复序列单元左右两测截取的序列长度，单位为 bp。 |
| [subconfiguration] | flanked\_sequence\_length (默认值 1000bp) | 在次要构型中，重复序列单元左右两测截取序列的长度，单位为 bp。 |
| [sequencing\_depth] | alignment software (默认值 minimap2) | 提供比对软件，可为 minimap2，bwa 或者 blast |
| evalue (默认值 1e-5) | blast 的参数，用于衡量匹配结果的显著性 |
| NGS\_single\_end | 二代单端数据文件位置，需单独提供 |
| NGS\_pair\_ends | 二代双端数据文件位置，需单独提供 |
| TGS | 三代测序数据文件位置，需单独提供 |
| TGS\_type | 设置三代测序平台类型，TGS 参数的补充参数，其值为 pacbio 或 ont |
| filter\_reads (默认值 Y) | 是否过滤测序read，过滤后可加快运行速度 |
| threads | 程序运行的线程数，为空时采用默认值 |
| [check\_spanning\_reads] | spanning\_read\_flanking\_repeat\_length (默认值 1 bp) | read 跨越重复序列的长度，为自然数 |
| spanning\_read\_number (默认值 1 bp) | 符合跨越重复序列的 read 的条数 |
| [refilter\_params] | refilter\_mode (默认值 N) | 是否再次过滤跨越重复序列的 read |
| refilter\_id | 再次过滤跨越重复序列的项目的编号 |
| spanning\_read\_flanking\_repeat\_length (默认值 5 bp) | 再次过滤跨越重复序列的 read 时，跨越重复序列的长度，为自然数 |
| spanning\_read\_number (默认值 5 bp) | 再次过滤跨越重复序列的read 时，符合跨越重复序列的 read 的条数 |

# 六、MiRI 运行结果的详解

以{project\_id}为名称的文件夹内存储着MiRI 运行后的所有结果。

文件夹final\_repeat-spanning\_results\_{project\_id}内存储着查询到的所有关于重复序列的信息：

① one\_chain\_without\_sufficient\_spanning\_reads.tsv

② one\_repeat\_unit\_without\_spanning\_reads.tsv

③ paired\_repeats\_for\_mapping.tsv

**④ paired\_repeats\_recomb-supporting\_ratio.tsv**

⑤ repeat\_sequences\_{project\_id}\_chr1.fasta

⑥ repeat\_sequences\_{project\_id}\_chr2.fasta

文件 ④ 为核心结果，文件 ③ 截取自文件 ④，可用于 MiRIV 软件绘制重组基因组图谱。文件 ① 存储的是DNA分子的两条链中，有一条链中的重复序列没有可跨越的read。文件 ② 存储的是DNA分子中，正负两条链中的重复序列均没有可跨越的read。文件 ⑤ ⑥ 是查询到的来自两条染色体的重复序列，为fasta格式。

文件夹subconfig\_repeat-spanned\_results\_{project\_id}存储的是read映射至次要构型的TSS上的中间结果。mainconfig\_repeat-spanned\_results\_{project\_id}存储的是read映射至主要构型的TSS上的中间结果。每一个文件夹存储一条TSS的映射结果，其文件夹的名字如下所示：

DR\_AD\_RPxxa\_RPxxb\_plus\_1000\_results

DR\_AD\_RPxxa\_RPxxb\_minus\_1000\_results

DR\_CB\_RPxxa\_RPxxe\_plus\_1000\_results

DR\_CB\_RPxxa\_RPxxe\_minus\_1000\_results

IR\_AC\_RPxxb\_RPxxc\_plus\_1000\_results

IR\_AC\_RPxxb\_RPxxc\_minus\_1000\_results

IR\_BD\_RPxxa\_RPxxb\_plus\_1000\_results

IR\_BD\_RPxxa\_RPxx1b\_minus\_1000\_results

DR\_AB\_RPxxa\_RPxxb\_plus\_1000\_results

DR\_AB\_RPxxa\_RPxxb\_minus\_1000\_results

DR\_CD\_RPxxa\_RPxxe\_plus\_1000\_results

DR\_CD\_RPxxa\_RPxxe\_minus\_1000\_results

IR\_AB\_RPxxa\_RPxxb\_plus\_1000\_results

IR\_AB\_RPxxa\_RPxxb\_minus\_1000\_results

IR\_CD\_RPxxa\_RPxxe\_plus\_1000\_results

IR\_CD\_RPxxa\_RPxxe\_minus\_1000\_results

文件夹名字各部分的命名规则如表 6-1 所示：

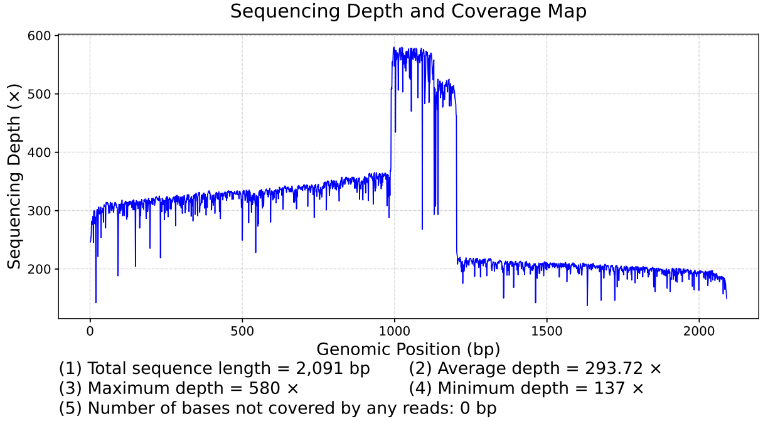
**表 6-1 结果文件夹名中各字符的含义**

|  |  |
| --- | --- |
| 符号 | 符号的含义 |
| DR | 正向重复序列 |
| IR | 反向重复序列 |
| AC | 次要构型中，以反向重复RPb为中心截取的TSS，见图2-3 |
| BD | 次要构型中，以反向重复RPa为中心截取的TSS，见图2-3 |
| AD | 次要构型中，以正向重复RPa为中心截取的TSS，见图2-3 |
| CB | 次要构型中，以正向重复RPb为中心截取的TSS，见图2-3 |
| AB | 主要构型中，以正向重复RPa为中心截取的TSS，见图2-3 |
| CD | 主要构型中，以正向重复RPb为中心截取的TSS，见图2-3 |
| RP | 表示重复序列 |
| xx | 表示重复序列的编号，为自然数 |
| a/b/c … | 同一重复序列的不同重复单元 |
| plus | 表示DNA的正链 |
| minus | 表示DNA的负链 |
| 1000 | 重复序列左右两侧截取的序列长度，即图2-3中的L |
| results | 文件夹名的后缀 |

每个文件夹内包含了TSS序列（fasta格式），跨越TSS中重复序列的read映射至TSS的测序深度，read映射至TSS的bam文档（图 6-1）。

**图 6-1 read映射至每一条TSS后的各种结果**

跨越TSS中重复序列的read映射至TSS的测序深度如图 6-2 所示，其测序深度的数值保存在coverage.txt中。bam文档可用Tablet等软件可视化read映射至TSS的实际情况，如图 6-3 所示。



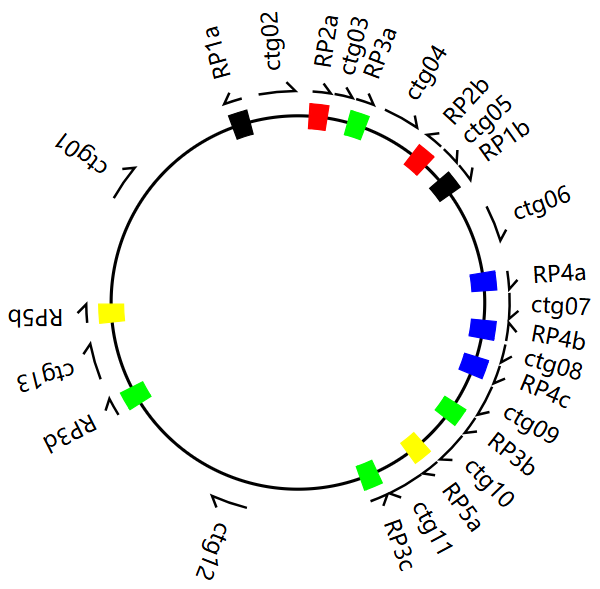
**图 6-2 跨越TSS中重复序列的read映射至TSS的测序深度**

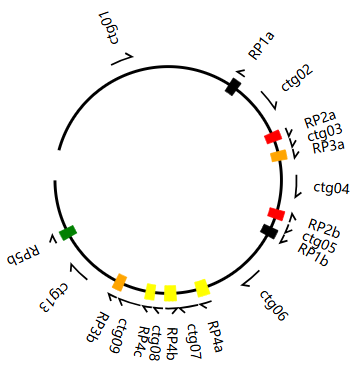


**图 6-3 跨越TSS中重复序列的read映射至TSS的可视化结果**

# 七、MiRIV 绘制图谱

软件MiRIV (**Mi**togenome **R**ecomb**i**nation **V**isulization) 可以从 MiRI 软件的结果出发，绘制环状基因组重组后的示意图，以展示由重复序列介导后线粒体基因组的各种subconfiguration的基因组图谱。图谱以环状表示，箭头表示DNA分子正链重组前后的走向，重复序列用带颜色的方块表示，相同颜色的方块表示同一重复序列的不同单元，不同的颜色表示不同的重复序列（图 7-1 A）。线性染色体的图谱为一个带缺口的环状图谱（图 7-1 B）。图谱半径的大小表示基因组序列的长短。





**B**

**A**

**图 7-1 MiRIV 绘制的基因组图谱示意图**

## 1、MiRIV 的运行

运行 MiRIV 的命令行如下：

python bin/MiRIV.py -c MiRIV.config.ini

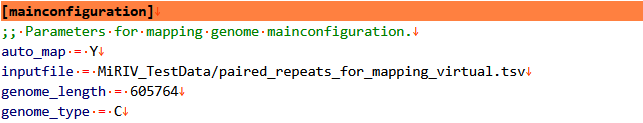
MiRIV 的参数以配置文件 .ini 的形式提供。绝大多数参数提供了默认值，仅有限几个参数需要用户提供。

## 2、MiRIV 的配置文件

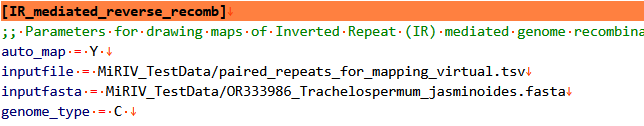
MiRIV可分别对主要构型（图7-2 A）、IR介导产生的次要构型（图7-2 B）和DR介导产生的次要构型（图7-2 C-E）绘制其基因组图谱。绘制图谱的时候，需要用户提供重复序列的位置信息，基因组的序列（fasta格式），基因组的长度以及基因组的类型（即线性还是环状结构）。

每一个模式均设置了 auto\_map 参数，用于控制是否运行（Y/N）相应的模式。auto\_map = N 时，MiRIV不运行相应的模式。auto\_map = Y，MiRIV 将自动绘制所有的基因组图谱。auto\_map = M，MiRIV 将在用户的指导下绘制用户指定的基因组图谱。

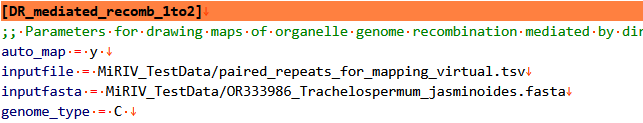
当遇到两条及多条染色体在DR介导下形成一条染色体的情况时，由于MiRIV每次只允许提供两条染色体，所以需要用户运行多次MiRIV，以将多条染色体重组为一条染色体。两条染色体重组为一条染色体的时候，其中一条染色体必须为环状结构（C），参数设置如图 7-2D 所示（chr1\_type、chr2\_type）。当两条染色体均为线性（L）的时候，MiRIV仅能实现两条染色体间序列的交叉重组，结果仍然是两条线性染色体，参数设置如图 7-2E 所示。



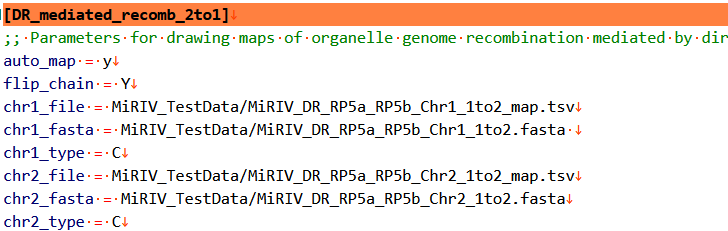
**A**



**B**

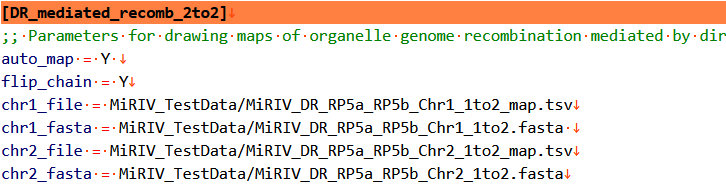


**C**



**D**

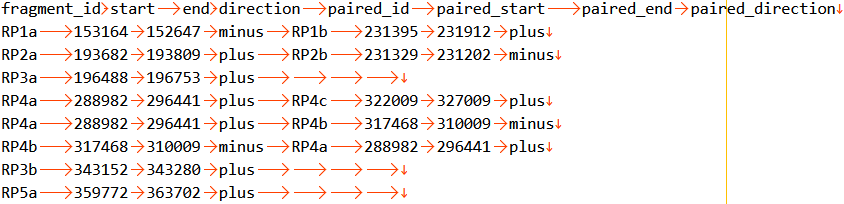
**E**



**图 7-2 MiRIV绘制各种图谱时所需的参数**

在 [DR\_mediated\_recomb\_2to2] 和 [DR\_mediated\_recomb\_2to1] 的两个模式中，两条染色体可以自由旋转，所以所有的重复单元之间都可以以正向重复介导基因组形成一条染色体，所以令 flip\_chain = Y，以允许所有的重复单元均以正向重复介导基因组形成一条染色体。

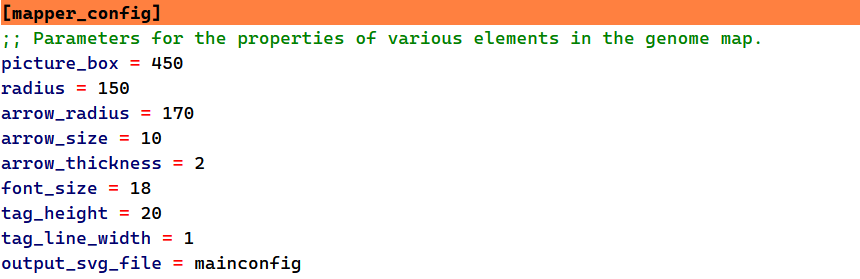
所有绘图模式中，inputfile、chr1\_file和chr2\_file的格式均为tsv格式的8列表，如图 7-3 所示。每一行表示一对重复序列，配对的重复序列可以缺省。表头和重复序列的名字必须与图 7-3 所示一致。



**图 7-3 绘图用8列表示意图（tsv格式）**

## 3、图谱中各元素的参数设置

[mapper\_config]选项用于设置绘制的基因组图谱中各元素的属性参数。各参数值如图 7-4 所示：



**图 7-4 [mapper\_config]各选项的设置**

各参数的默认值及其含义如表 7-1 所示：

**表 7-1 [mapper\_config]各选项的默认值及其含义**

|  |  |
| --- | --- |
| 参数 | 参数的取值 |
| picture\_box | 输出图像的大小（正方形一边的长度），\*\*默认值=280\*\* |
| radius | 基因组图谱的半径，决定图片的大小，\*\*默认=150\*\* |
| arrow\_radius | 箭头所在圆的半径，\*\*默认=170\*\* |
| arrow\_size | 箭头的大小，\*\*默认=10\*\* |
| arrow\_thickness | 箭头线的粗细，\*\*Default=2\*\* |
| font\_size | 字体大小，\*\*Default=18\*\* |
| tag\_height | 标签（环形扇形）的高度，\*\*Default=20\*\* |
| tag\_line\_width | 标签（环形扇形）轮廓厚度，\*\*默认=1\*\* |

## 4、重复序列的颜色参数

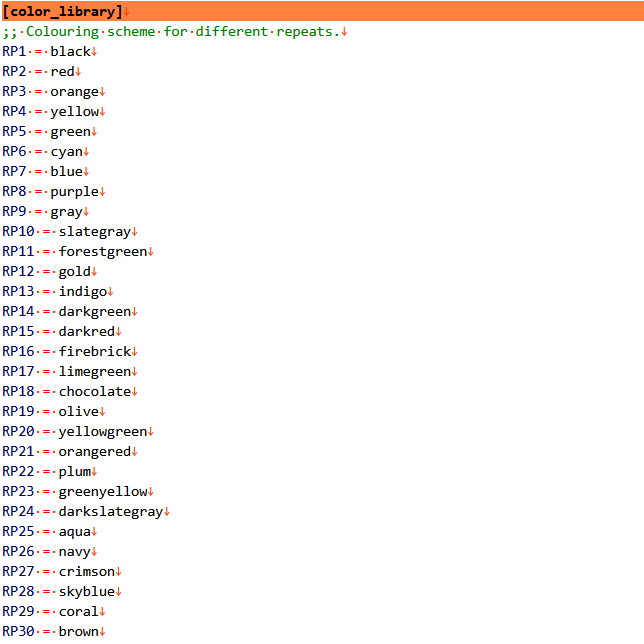
[color\_library]选项用于设置基因组图谱中重复序列单元的颜色。颜色可用 60 种内置颜色的英语单词表示，也可以用RGB和16进制表示。

内置的60种颜色筛选自python的webcolors库，分别为：black，red，orange，yellow，green，cyan，blue，purple，brown，gray，darkslategray，dimgray，navy，indigo，darkgreen，darkred，firebrick，crimson，chocolate，olive，yellowgreen，lawngreen，limegreen，greenyellow，lightseagreen，seagreen，darkseagreen，lightgreen，forestgreen，darkcyan，mediumturquoise，turquoise，aquamarine，mediumaquamarine，aqua，deepskyblue，skyblue，steelblue，cadetblue，royalblue，mediumblue，darkviolet，plum，deeppink，hotpink，pink，palevioletred，mediumvioletred，coral，orangered，darkorange，goldenrod，gold，khaki，darkkhaki，wheat，lightgrey，lightslategray，slategray，darkgray.

RGB 的颜色设置如：0.0.0（黑），255.0.0（赤），255.165.0（橙），255.255.0（黄），0.255.0（绿），0.255.255（青），0.0.255（蓝），128.0.128（紫）。

16 进制表示的颜色如：#000000（黑），#FF0000（赤），#FFA500（橙），#FFFF00（黄），#00FF00（绿），#00FFFF（青），#0000FF（蓝），#800080（紫）。

MiRIV最多允许30个重复单元同时上色，具体的设置如图 7-5 所示：



**图 7-5 基因组图谱内重复序列上色方案**

## 5、多个图谱排版的参数

[Arrange\_map]用于设置多个图谱排版的参数，图谱按照九宫格排版（图 7-6 A）：

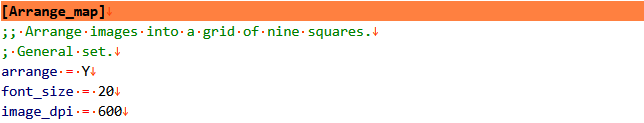
arrange：是否要对多个图谱进行排版。Yes/Y表示进行排版，No/N表示不要排版。

font\_size: 简单标签的字体的大小。

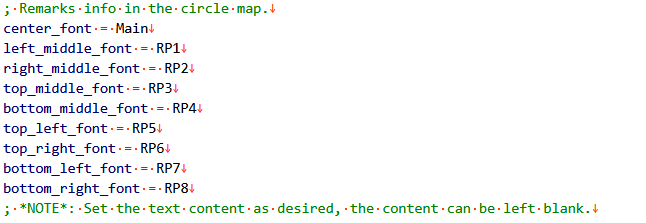
image\_dpi：图谱排版后图片的分辨率。

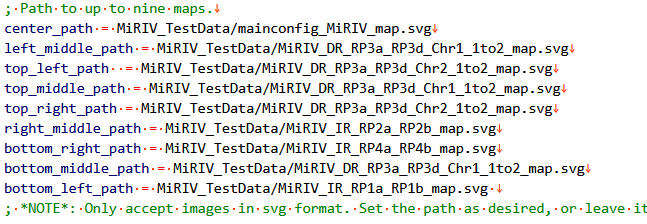
每个图谱可以在图谱的中间位置设置简单标签（图 7-6 B）。并且每个图谱可以独立设置在排版后图片中的位置（图 7-6 C）。设置的标签可由大小写字母、数字和下划线组成。

**A**



**B**

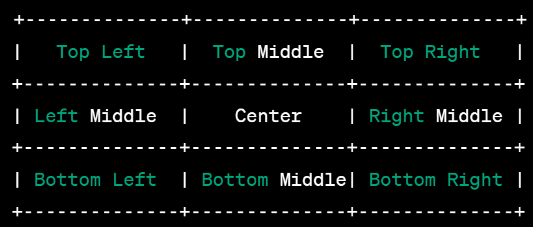




**C**

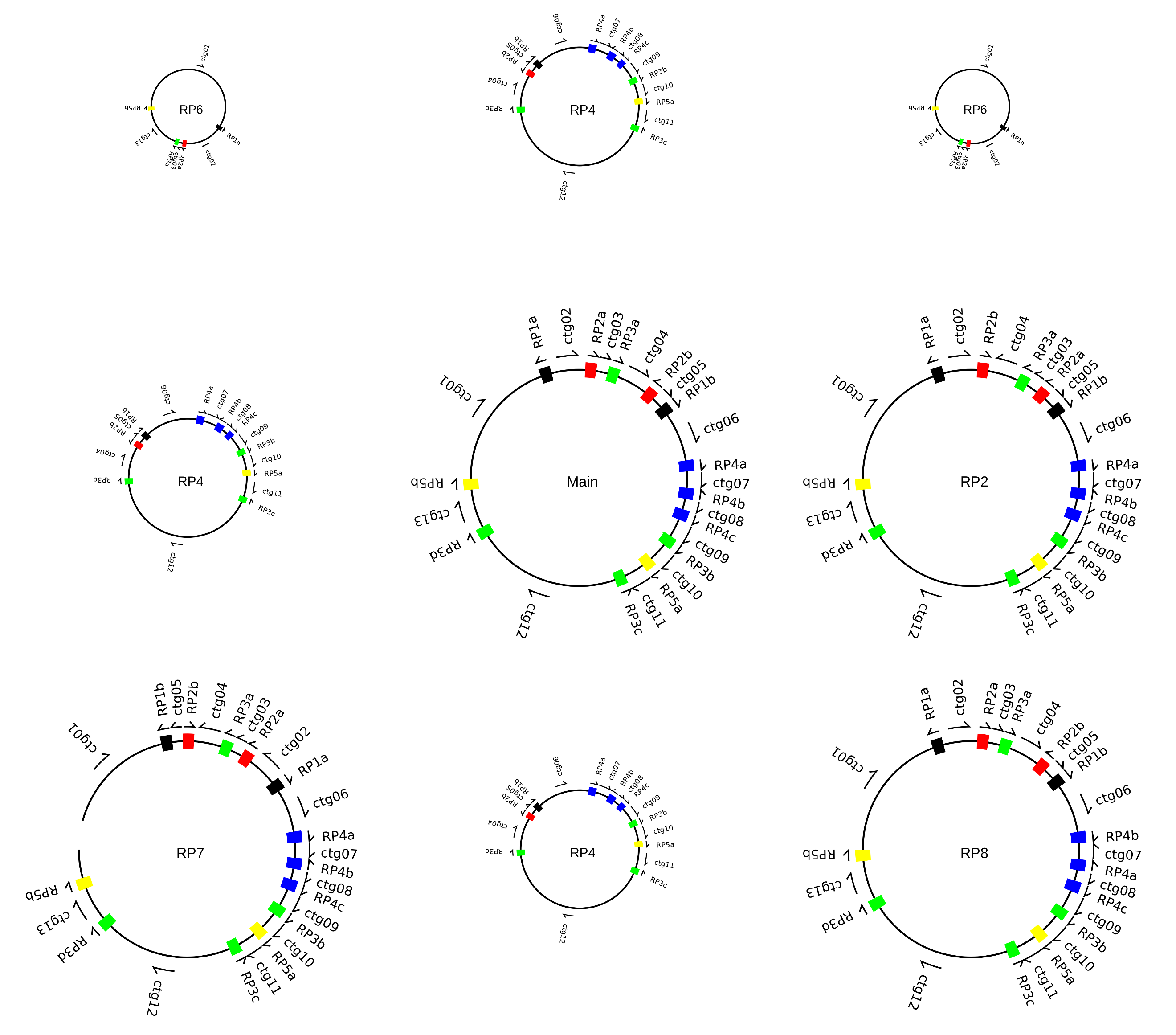
**图 7-6 多个图谱排版的参数设置**

图谱在九宫格内的位置如下图 7-7 所示：



**图 7-7 图谱在九宫格内的位置示意图**

图谱在九宫格内排版后的效果如图 7-8 所示：



**图 7-8 九宫格排版后的效果图**

## 6、各模式绘图结果解读

[mainconfiguration]、[IR\_mediated\_reverse\_recomb]、[DR\_mediated\_recomb\_1to2]、[DR\_mediated\_recomb\_2to1]、[DR\_mediated\_recomb\_2to2]、[Arrange\_map]几个模式运行后的结果存储在以{project\_id}为名字的文件夹内：

[mainconfiguration]：mainconfig\_{project\_id}

[IR\_mediated\_reverse\_recomb]：Inv\_Rev\_{project\_id}

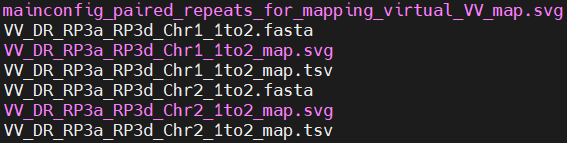
[DR\_mediated\_recomb\_1to2]：DR\_1to2\_{project\_id}

[DR\_mediated\_recomb\_2to1]：DR\_2to1\_{project\_id}

[DR\_mediated\_recomb\_2to2]：DR\_2to2\_{project\_id}

[Arrange\_map]：map\_nine\_squares\_{project\_id}

以 [DR\_mediated\_recomb\_1to2] 的结果为例，结果有三种类型：一是主要构型的重复序列的图谱，它是根据用户输入的8列表（8CT.tsv）产生，svg格式，可用网页浏览器和Adobe Illustrator CS6等软件打开；二是输入的序列产生的第一条染色体（chr1），共三个文件，分别是fasta格式的序列，svg格式的重复序列的图谱以及图谱对应的8列表；三是输入的序列产生的第二条染色体（chr2）的结果文件。

其他模式的输出结果类似于 [DR\_mediated\_recomb\_1to2] 的结果。其中，[DR\_mediated\_recomb\_2to1] 模式中需要将三条及以上条序列转换为一条染色体时，前一步结果中的8列表（tsv格式）和fasta格式的文件可作为下一次MiRIV的输入文件。

**图 7-9 各模式输出的结果的示意图**