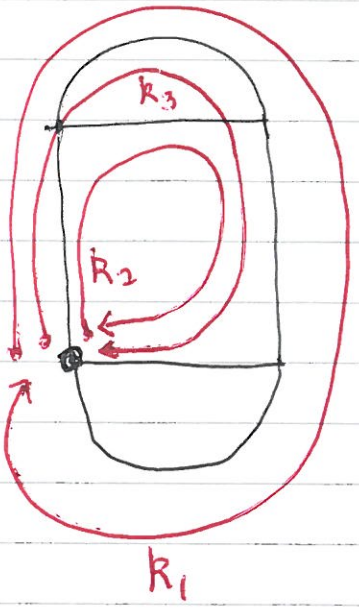
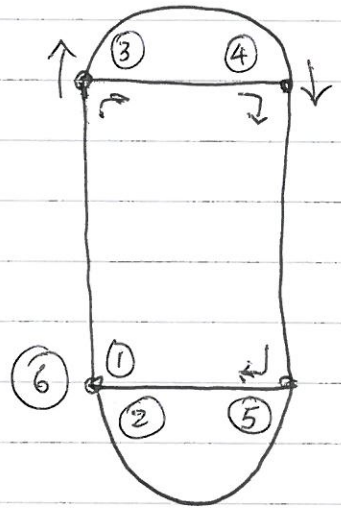
Generators





Pairs to be resolved:

1 = {(1,2), (1,3), (2,2), (2,3), (3,3)}

2 = {(1,1), (1,2), (1,3)}

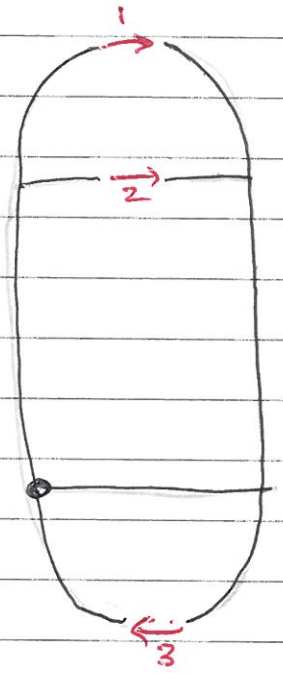
3 = {(1,2), (2,3)}

4 = {(1,2), (2,3)}

5 = {(1,2), (1,3)}

6 = {(1,1), (1,2), (1,3)}

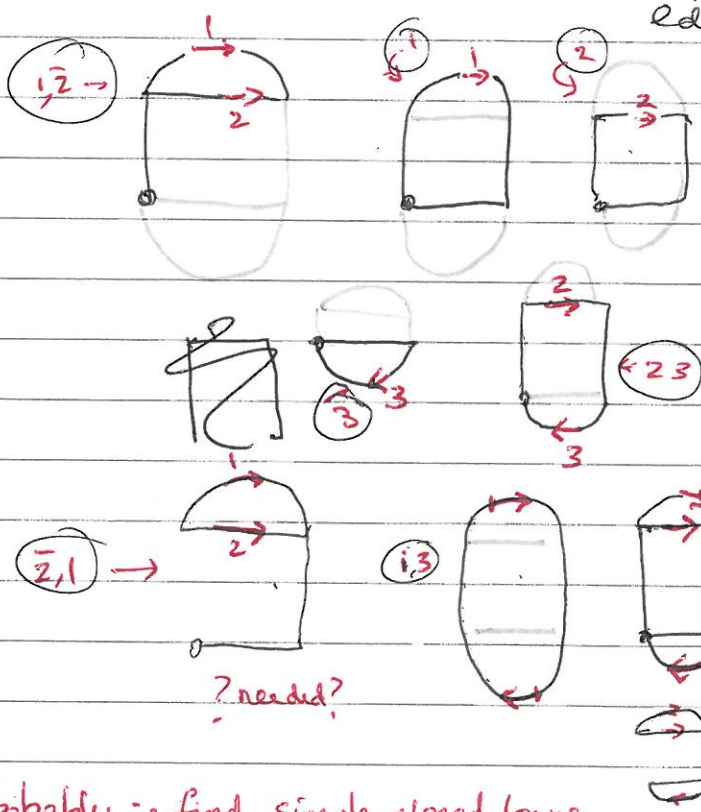
Loops to check



Max^l subtree of "underlying" graph

Edges (directed) outside max^l subtree

loops : ∘ —Tree— → —Tree— ⋯ ∘ with no repeated edges

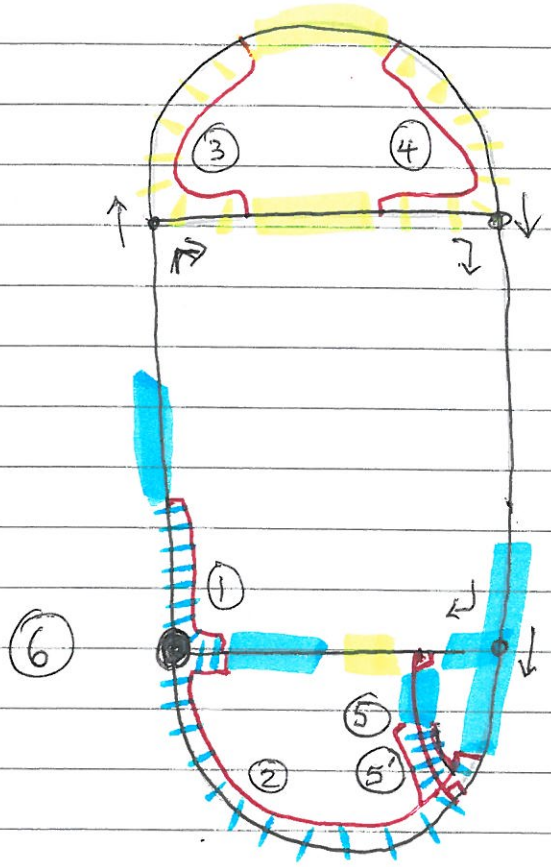


[Probably : • find simple closed loops
• paths from base pt to each one
• unions (+ differences of these)]

9 loops in total. (some not strictly necessary)

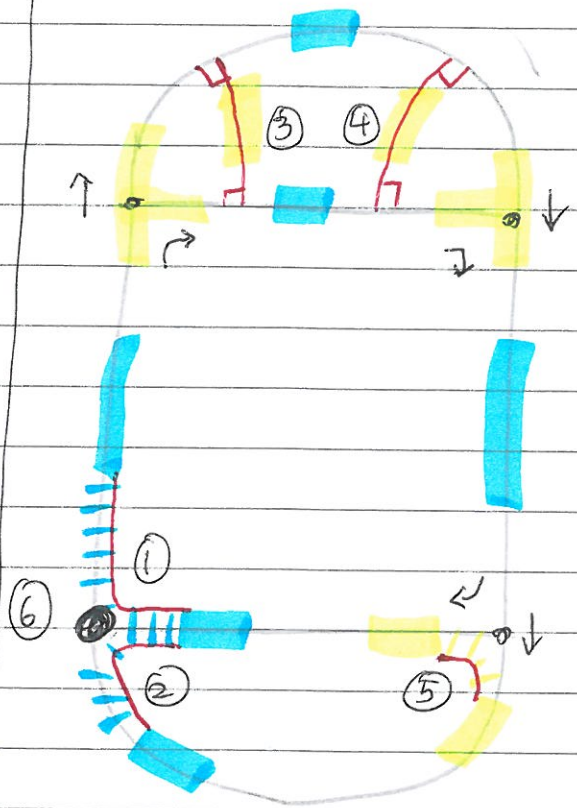
In exx's 1 & 2 above: all loops in normal form.

Example 1



Resolve ① ··· ⑤ → ⑥ comes for free.
Next need to add (5') to allow
normal form of ↻ repeated.

Example 2



Resolve ① --> ⑤ ; ⑥ comes free.

All loops (& iterations of loops) have readable normal forms.

Key

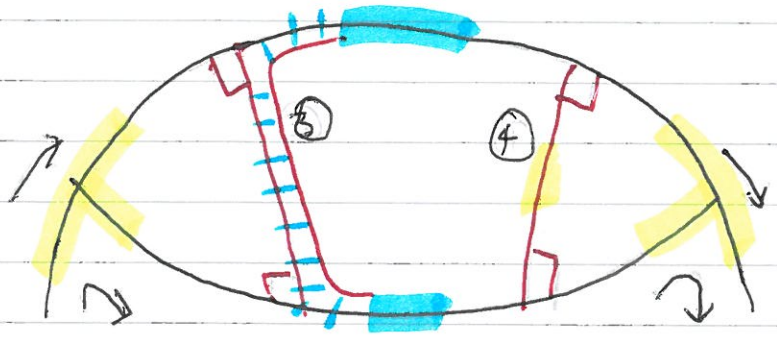
original X_1, X_2 edges

orig Z edges

New X_1, X_2 Z edges

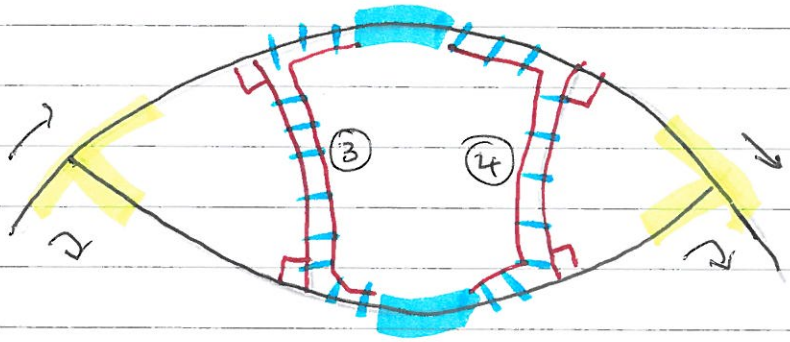
Z's with newly added X_1 or X_2 edges

Exx 2 (3) collapses to Zs



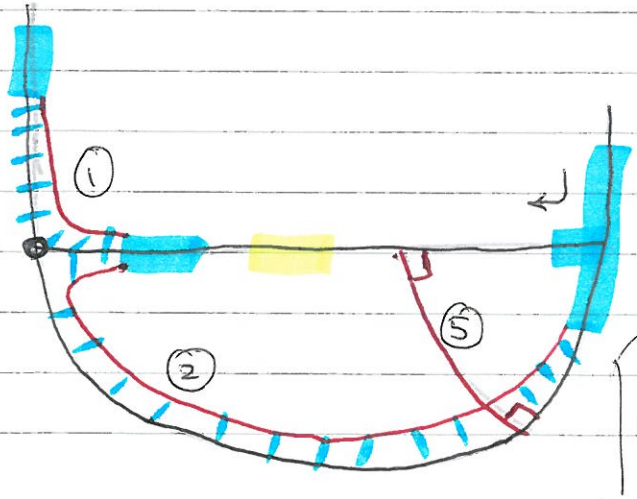
& all is well.

Exx 2 (3) & (4) collapse to Zs

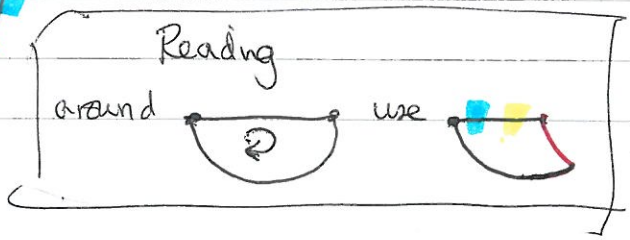


All's well!

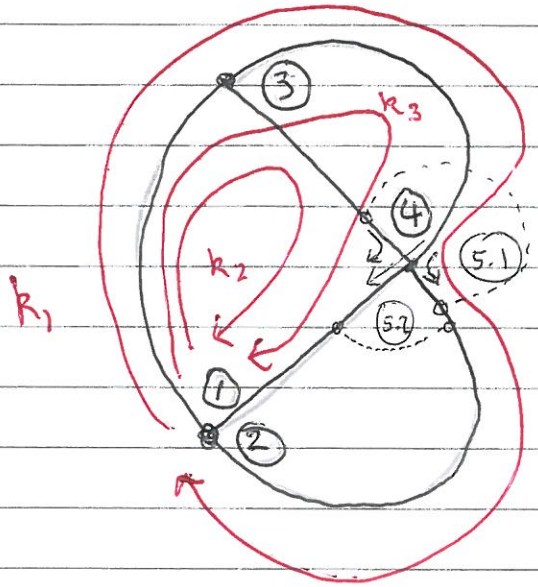
Exx 1 (5) collapses to Z's



All ok.



Degenerate case of  where right hand vertical edge shrinks to a vertex:



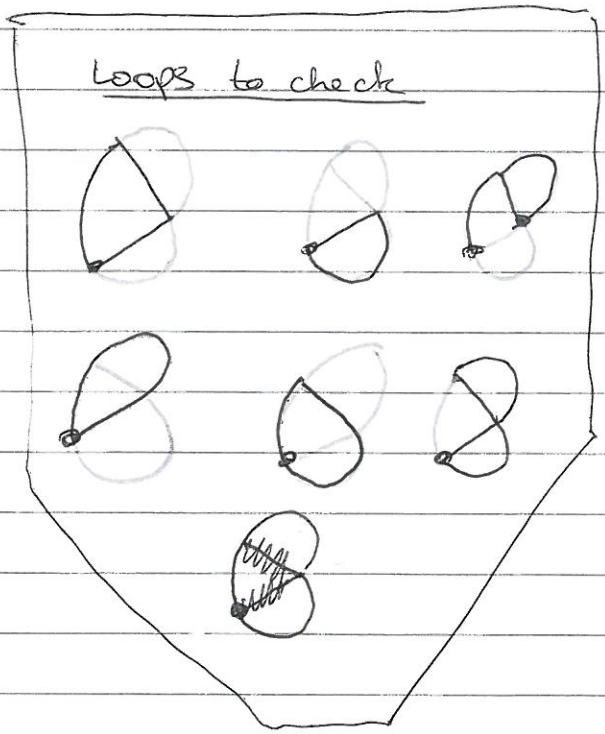
①, ② ③ as before

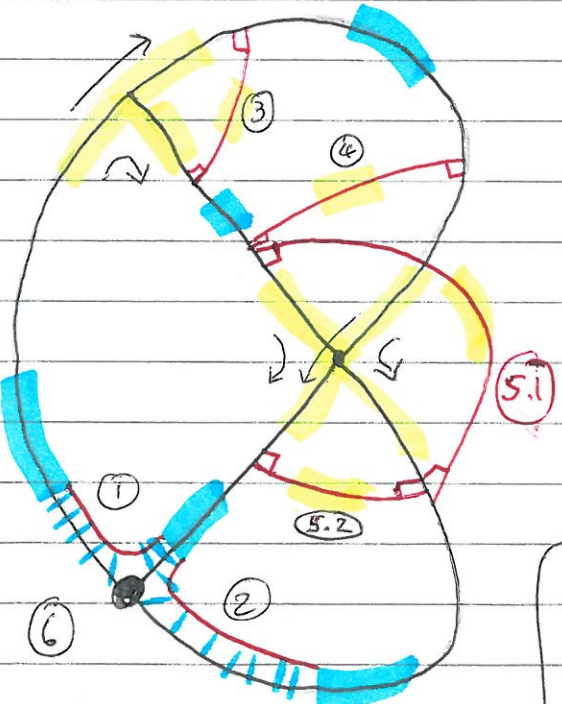
④ ~~1~~ as before, ⑥ as before

5.1 $= \{(1,2)\}$ ~~F~~

5.2 $= \{(1,2), (1,3)\}$

Loops to check





All ok.

If ③ or ④ degenerate to Z's
- as before

If 5.1 or 5.2 degen to Z's
- add app^{r} ships.