

Alternating Inference Chains

From sudokuwiki.org, the puzzle solver's site

| | | |
|---|---|---|
| 2 | | |
| | 3 | 6 |
| 5 | | 7 |

Chaining strategies now take a new leap with **Alternating Inference Chains**. These extend [X-Cycle](#) into a new dimension - where X-Cycles stuck to a single number, AICs use any candidate number.

AICs encapsulate all the discussion of chaining strategies so far. It's very useful to split out chain-like strategies into X-Wings, XY-Chains, Forcing Chains, XYZ-Wings, X-Cycles, Nice Loops and so on, since they have special characteristics which make them spottable. But it turns out they are all part of a larger extended family.

As we saw in the [previous chapter](#), alternation is just what X-Cycles are about. However, you'll remember that X-Cycles are applied only to a single candidate number. AICs, on the other hand, take everything from an X-Cycle and extend the logic to as many different candidate numbers as necessary.

AICs ask the question "How many ways are there to make a strong or a weak link?" If there is more than one way, we can join them up in an alternating manner and make deductions leading to eliminations. Let's look back on the previous chain-like strategies and note the following:

- We can link two candidates of the same value in a unit - this is called "bi-location" (X-Cycles).
- We can link two different candidates in the same cell - this is called "bi-value".

There are also other ways (see later articles), but for now let's keep it simple and stick to these two dimensions - links between cells and within cells.

Nice Loops Rule 1

Nice Loops that alternate all the way round are said to be 'continuous', and they must have an even number of nodes. With a continuous AIC, candidates are not removed from the loop since the loop does not have any flaws. Instead we are looking to eliminate on the units that can be seen by two or more cells that belong to the loop.

Specifically, if a unit has an ON number X and an **OFF** number X then one or other will be the solution. All other candidates X in that unit can be removed. These are called *off-chain* eliminations. Take this example:

AIC (Alternating Inference Chain) Rule 1:

+4[B7]-4[B2]+7[B2]-7[B5]+6[B5]

-6[H5]+4[H5]-4[H7]+4[B7]

- Off-chain candidate 7 taken off B3 - weak link: B2 to B5
- Off-chain candidate 6 taken off J5 - weak link: B5 to H5

Two off-chain eliminations occur on the weak links **B2** to **B5** (candidate 7) and **B5** to **H5** (candidate 6).

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|----------|------------|-------------|---------------|-------------|------------------|-------------|---------------|-------------|
| A | 1 7 9 | 3 | 8 | 4 | 2 7 | 1 2 5 | 2 5 9 | 1 5 6 9 | 2 6 9 |
| B | 2 | 4 7 | 1 7 | 9 | 6 7 | 1 5 6 | 4 5 | 3 | 8 |
| C | 1 4 9 | 6 | 5 | 3 | 8 | 1 2 | 7 | 1 9 | 2 4 9 |
| D | 5 7 8 | 5 8 9 | 2 7 9 | 2 6 7 8 | 3 | 2 6 9 | 1 | 4 | 2 6 9 |
| E | 6 | 4 7 8 9 | 3 | 2 7 8 | 1 | 2 4 9 | 2 8 9 | 7 9 | 5 |
| F | 4 7 8 | 1 | 2 7 9 | 2 6 7 8 | 5 | 2 4 6 9 | 2 8 9 | 6 7 9 | 3 |
| G | 7 8 | 7 8 | 4 | 5 | 9 | 3 | 6 | 2 | 1 |
| H | 3 | 2 | 1 6 9 | 1 6 | 4 6 | 8 | 4 5 9 | 5 9 | 7 |
| J | 1 5 | 5 9 | 1 6 9 | 1 2 6 | 2 4 6 | 7 | 3 | 8 | 4 9 |

AIC Rule 1 : *Load Example* or : *From the Start*

This is a classic and pleasingly short Continuous Alternating Inference Chain.

Starting a **B7** we turn 4 ON.

This removes the 4 in **B2** turning on 7 in that cell.

That turns **OFF** 7 in **B5** turning on 6.

6 in **B5** turns off 6 in **H5**.

That turns on 4 in **H5** removing 4 in **H7**

Which confirms 4 must be ON in **B7**

Thus...there is no contradiction in the loop. The nice thing about Nice Loops is they can be reversed. Try starting with 4 ON in **B7** and turning 4 OFF in **H7** - it will come back round with the same conclusion. In fact, the loop is especially "Nice" because you can start with any candidate in the loop and work your way round, provided it is the same On/Off state as described in the example.

So having proved the loop we can look for extra candidates on any unit linked by the chain - or indeed, extra candidates in the same cell where an ON/OFF has occurred. (There are none in this case, only bi-value cells have been used).

Off-chain eliminations can also occur within a cell. In this excellent example sent to me by **Andreas von Delius**, there is a fine Nice Loop. It can start anywhere and can be traced in either direction, but the solver picks it up on **A3**. The loop picks out a series of 6s and 2s that alternate on and off. All the links between cells are strong links, but there are two switches between 2 and 6 in the loop. In those cells, **A8** and **H1**, other candidates exist. 2 or 6 will be the answer - we don't know which yet, but one or other, so other candidates in those cells can be removed.

AIC (Alternating Inference Chain) Rule 1:

$+6[A3]-6[C1]+6[H1]-2[H1]$

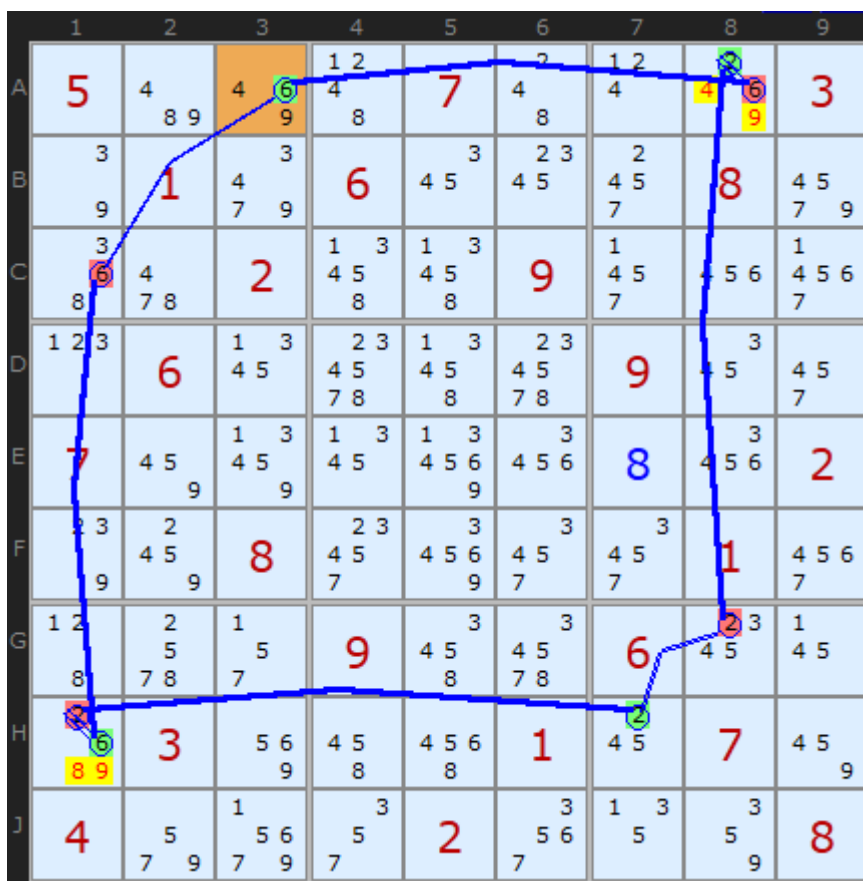
$+2[H7]-2[G8]+2[A8]-6[A8]+6[A3]$

- Off-chain candidates 8/9 taken off cell

H1, link is between 6 and 2 in H1

- Off-chain candidates 4/9 taken off cell A8,
link is between 2 and 6 in A8

Nice Loops Rule 2



Off-chain eliminations in cells : [Load Example](#) or : [From the Start](#)

Now we turn to flawed loops - ones that show a discontinuity. In terms of strong/weak links, there are two types, as described in the article on [X-Cycles](#). Those where two strong links join up and those where two weak links join up.

If the adjacent links are links with strong inference (solid line), a candidate can be fixed in the cell at the discontinuity. It removes all other candidates as is the solution to that cell. This type is unfortunately much rarer than the Nice Loop Rule 3, two weak links.

This AIC starts and ends in [A2](#). Setting 5 to be **OFF** creates a loop and a contradiction - that 5 must be ON if 5 is **OFF**. So removing the 5 forces it to reappear! We can't safely remove the 5 in A2 so we must set it - removing the other candidates.

AIC on 5 (Discontinuous Alternating Nice Loop, length 8):

$-5[A2]+5[G2]-3[G2]+3[G8]$

$-3[B8]+3[A9]-5[A9]+5[A2]$

- Contradiction: When 5 is removed from A2 the chain implies it must be 5 - other candidates 1/8 can be removed

In terms of strong/weak links, we have a candidate where two strong link join up, hence the thick lines drawn by the solver.

Nice Loops Rule 3

[illegible]

AIC Rule 2: *Load Example* or: *From the Start*

Our third rule dictates what happens when two weak links form a discontinuity in a loop:

If the adjacent links are links with weak inference (broken line), a candidate can be eliminated from the cell at the discontinuity. In terms of ON/**OFF** this is where you try and set a candidate to be ON but the loop comes round and shows that doing so forces that candidate to be turned **OFF**.

A2 is again the start and end of the loop (coincidence, its a completely different puzzle).

By setting A2 to be 5 (+5[A2]) we turn off 5 in **A7**

That leaves 7 in **A7** turning off 7 in **F7**.

The strong (bi-location) link makes 7 on in **F2**.

That removes 7 in E2 making 5 the solution in that cell.

If 5 is ON in **E2** it has to be **OFF** in **A2**....

Contradiction!

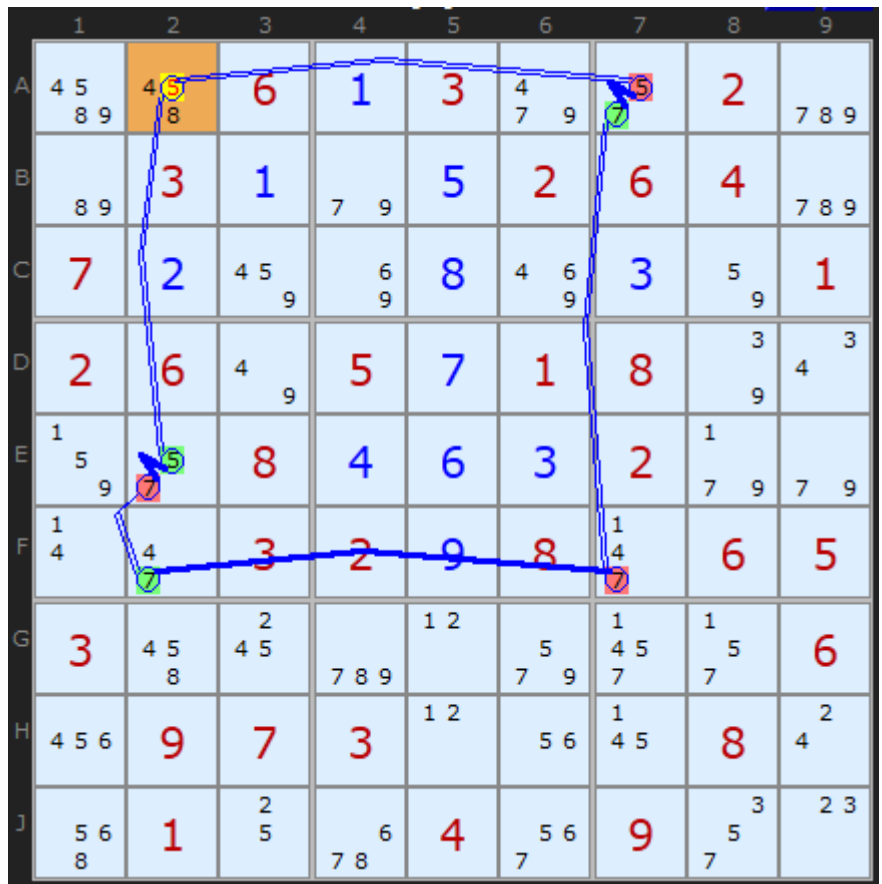
AIC on 5 (Discontinuous Alternating Nice Loop, length 8):

$$+5[A_2]-5[A_7]+7[A_7]-7[F_7]$$

$$+7[F2]-7[E2]+5[E2]-5[A2]$$

- Contradiction: When A2 is set to 5 the chain implies it cannot be 5 - it can be removed

In summary...



AIC Rule 3 : [Load Example](#) or : [From the Start](#)

AICs are chains of links going across a unit and within a cell.

If a candidate is turned ON you create a weak link turning **OFF** any other candidates in that cell and across the units it can see.

If you turn a candidate **OFF** you can turn ON other candidates in that cell (if there are only two candidates in the cell - **bi-value**) or across the unit if candidate X occurs just twice (bi-location).

However, bi-value and bi-location are just two ways of making chain links. There are other interesting ways of making chain links: [Grouped Cells](#) and [Almost Locked Sets](#) are documented here and other patterns as well can be made into links, even [Unique Rectangles](#).

AIC Exemplars

These puzzles require an Alternating Inference Chain strategy at some point but are otherwise trivial.

They make good practice puzzles.

- [Exemplar 1, x1 \(score 112\)](#)
- [Exemplar 2, x1 \(score 112\)](#)
- [Exemplar 3, x1 \(score 116\)](#)
- [Exemplar 4, x1 \(score 122\)](#)
- [Exemplar 5, x1 \(score 142\)](#)

All are non-symmetrical but valid Sudoku puzzles.
Thanks for Klaus Brenner for these examples

Go back to [Grouped X-Cycles](#) Continue to [AICs with Grouped Cells](#)

| | | |
|---|---|---|
| 2 | | |
| | 3 | 6 |
| 5 | | 7 |
