**Introduction**

Let’s start describing the parameters for the heuristics, we have two categories of parameters:

1. The variable-value ordering heuristics, determining the way the variables are chosen during the search, we have:
   1. Input Order: The variables are chosen in order of input.
   2. First Fail: The variables are chosen based on the size of the domain, smallest size first.
   3. Weighted Degree: The variables are chosen based on quota of the of the domain size over weighted degree, implying that the variables with small domain and large weighted degree are chosen first.
2. Constraints on the variables:
   1. Minimum Value: Constraining the variables to take the smallest value in the domain.
   2. Random Value: Constraining the variables to take a random value from the domain.

**N-Queens**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **input\_order** | | **first\_fail** | | **dom\_w\_deg** | |
|  | **indomain\_min** | **indomain\_random** | **indomain\_min** | **indomain\_random** | **indomain\_min** | **indomain\_random** |
| **30x30** | 1,588,827 | 9 | 15 | **1** | 15 | **1** |
| **35x35** | 2,828,740 | 10 | 21 | **0** | 21 | **0** |
| **45x45** | - | 6 | 6 | **1** | 6 | **1** |
| **50x50** | - | 42 | 123 | **10** | 123 | **10** |

As we can see, **input\_order+indomain\_min** heuristics give the worst results, this is probably due to the fact that using the lowest value in the domain for the variables selected in the order of the input gives the solver no choice in the selection of the better variables to assign first, it also gives no choice in the values assigned to them, this implies that if, let’s suppose, the first choice is wrong and leads to no solutions, the solver is going to spend a lot of time and fail many times before exiting the subtree of the first decision. That’s probably the reason of the 45x45 and 50x50 instances not terminating at all (in 5 minutes).

This is also magnified through the results of **input\_order+**

**Poster**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ***input\_order*** | | ***first\_fail*** | | ***dom\_w\_deg*** | |
|  | ***indomain\_min*** | ***indomain\_random*** | ***indomain\_min*** | ***indomain\_random*** | ***indomain\_min*** | ***indomain\_random*** |
| **19x19** | 1,362,457 fails  6.369s | - | **239,954 fails**  **1.385s** | 2,929,153 fails  13.360s | **236,024 fails**  **1.257s** | 2,929,030 fails  13.382s |
| **20x20** | - | - | **1,873 fails**  **120ms** | 5,797,312 fails  25.527s | **1,873 fails**  **129ms** | 5,797,456 fails  25.716s |

**Ordered Poster**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ***input\_order*** | | ***first\_fail*** | | ***dom\_w\_deg*** | |
|  | ***indomain\_min*** | ***indomain\_random*** | ***indomain\_min*** | ***indomain\_random*** | ***indomain\_min*** | ***indomain\_random*** |
| **19x19** | 30 fails  112ms | - | **252,210 fails**  **1.326s** | 3,637,566 fails  16.626s | **245,441 fails**  **1.313s** | 3,457,753 fails  15.849s |
| **20x20** | 323 fails  116ms | - | **1,737 fails**  **120ms** | 4,402,830 fails  19.69s | **1,737 fails**  **121ms** | 4,402,770 fails  19.197s |

**QuasiGroup Completation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ***default\_search*** | | ***dom\_w\_deg + random*** | | ***dom\_w\_deg + random + restart*** |
| **30-03** | - | | 1,061,184 fails | 1.20s |  |
| **30-05** | 657,955 fails | 54.687s | 5,885 fails | 690ms |  |
| **30-08** | 627 fails | 186ms | 6,403 fails | 738ms |  |
| **30-12** | 259,082 fails | 19.690s | 53,200 fails | 4.552s |  |
| **30-19** | 381,330 fails | 33.58s | **-** | |  |