# N-QUEENS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***N*** | ***#sols*** | ***R*** | ***RC1*** | ***RC2*** | ***RC3*** | ***alldiff*** |
| **8** | 92 | 891 | 500 | 593 | 864 | 254 |
| **9** | 352 | 4,262 | 2,656 | 2,772 | 4,603 | 849 |
| **10** | 724 | 23,291 | 13,996 | 13,593 | 23,195 | 3,722 |
| **12** | 14,200 | 773,550 | 355,041 | 380,595 | 820,127 | 75,823 |

|  |  |  |
| --- | --- | --- |
| ***N*** | ***#sols*** | ***alldiffsym*** |
| **8** | 12 | 75 |
| **9** | 46 | 246 |
| **10** | 92 | 871 |
| **12** | 1,787 | 16,443 |

*What is happening when going r → rc1 → alldiff ? Why?*

These tables contain the solutions and failures for the N-queens problem of six models and are tested with four different values of N.

Looking at the data in the table, it is possible to notice that the solutions and failures vary based on the value of N which gradually increases.

The Row Model has the highest number of failures which are due to the presence of only one constraint for the column attachment and only one for the diagonal attachment. A fundamental part of the code is the "channeling constraint" which allows you to reduce errors and obtain a more reliable solution.

The RC1 Combined Model also presents numerous failures compared to the Alldiff model which is the one with the least number of failures.

It is noted that the Alldiff model presents fewer failures: this is justified by the fact that it uses three global constraints which allow to obtain a better result.

*What is happening when going rc1 → rc2 → rc3 ? Why?*

The RC1 model is the one that provides the most constraints of all three combined models.

However, if we look at the RC1, RC2 and RC3 models, we notice if there are numerous constraints we get fewer failures when running the code. If, however, we remove constraints from the model (as in the case of RC3), we obtain a greater number of failures compared to the RC1 and RC2 models.

Between the first two models we have an approximately equal number of failures; there is a greater difference with the RC3 model.

These differences are justified by the constraints that are removed by moving from the RC1 model to the RC2 model in which the alldifferent() constraints are removed. By moving from the RC2 model to the RC3, a constraint relating to the diagonal attachment for the columns is removed.

The removal of all these constraints, even if the remaining ones allow a correct solution, determine a reduction in performance.

*What is happening when going alldiff → alldiffsym? Why?*

Also referring to the number of failures of both models, it can be noted that the "alldiffsym" model has a lower number of solutions than the "alldiff" model.

This is due to the fact that the "alldiffsym" model uses additional constraints which allowed reducing the number of valid solutions, reducing the symmetries in the solutions.

**Warning**: we use the terminology "failure" in a search tree, not "error".

Well done! However, you need to be more specific when you comment on the combined models.  Currently, you do not explain the reason why:

* "A fundamental part of the code is the "channeling constraint" which allows you to reduce errors and obtain a more reliable solution."
* "The removal of all these constraints, even if the remaining ones allow a correct solution, determine a reduction in performance."

I suggest that you pass over the modelling topics and give concrete reasons why a combined model (like rc1) is better than r, and why removing the constraints when going from rc1 to rc2 and then to rc3 degrade the performance.

**RISPOSTA**

The combined RC1 model is better than R because it considers both constraints on rows and columns and because it integrates a second model that broadens the domain of variables involved, providing a more complete representation of the problem. The integration of a second model which broadens the domain of the variables involved allows for increasing propagation and modifying the search space.

Models RC2 and RC3 degrade performance compared to RC1 because they remove or neglect important constraints that affect the correctness of solutions or the ability to find them efficiently.

In the RC2 model, two alleys are removed: the one on the rows and the one on the columns. This is possible thanks to the presence of the channeling constraint which provides the same information that the Xi and Yj values must be different. Even if the information is the same, the presence of two implicit constraints improves the performance of the model.

The RC3 model performs worse than RC2 because we remove another implicit constraint that provides extra information for the diagonal attack.

**Comments on the second submission:**

**- R to RC1:**" broadens the domain of variables involved" -> I don't know what you mean by this, I don't remember to have said anything like that 😅 Combining models in this case benefits from "enhanced constraint propagation". Do you understand what we mean by this?

*La frase "amplia il dominio delle variabili coinvolte" potrebbe essere un po' ambigua. Potreste voler spiegare meglio cosa intendete con "ampliare il dominio delle variabili coinvolte", ad esempio, specificando che il secondo modello introduce nuovi vincoli o variabili che migliorano la rappresentazione del problema.*

Il dominio in realtà viene ridotto, quindi il problema è più facile da risolvere.

"propagazione dei vincoli migliorata" è un concetto chiave, ma spiegarlo in modo più dettagliato. Potreste spiegare come l'uso di due modelli che contengono vincoli impliciti crea una propagazione dei vincoli più forte, aiutando così il risolutore a trovare soluzioni più efficientemente.

Con il RC1 abbiamo aggiunto il channelling constraint che permette di mantenere la consistenza tra le variabili dei due modelli. I benefici che comporta il miglioramento della propagazione.

PRIMA PARTE

Sorry, maybe we did not explain ourselves well. By "broadens the domain of variables involved" we meant that the RC1 model introduces a new variable (cols) and consequently new constraints. The introduction of new constraints broadens the domain and this allows an easier resolution of the problem.

SECONDA PARTE

In the combined RC1 model, we added Channeling Constraints, which plays an important role: it allows us to maintain consistency between the variables in the two models. The combined model has several benefits, including improved propagation.

Thanks to Channeling Constraints, we are able to use constraints to maintain consistency between the variables in the two models: these additional constraints help identify good solutions more efficiently, since the constraints help "propagate" information more quickly between the variables involved.

**- R1 to R2 to RC3:**none of the removed constraints affect the correctness. As I explained in the lecture (which you can verify by studying again), the removed constraints are implied by the other constraints (alldifferent by the channeling, column attack via the same constraint in the other model + channelling,  that's why they are safe to remove. Does this make sense?

What changes however is the solver performance as you also said. The presence of the implied constraints lets the solver find the solutions quicker (as you also noticed in the sequence problem). Indeed, you wrote for the sequence puzzle the following: "The constraints inserted only provide additional information and create greater propagation for the model which allows it to arrive at the solution more quickly."

We explained ourselves wrongly. Removing the constraints does not affect correctness because the constraints are implicitly present in the others. In fact, as we wrote for the sequence puzzle, the implicit constraints only give extra information and allow more propagation for the model: this allows us to arrive at the solution more quickly. Removing the constraints does not lead to erroneous results but only to a small drop in performance.

**RISPOSTA**

Sorry, maybe we did not explain ourselves well. By "broadens the domain of variables involved" we meant that the RC1 model introduces a new variable (cols) and consequently new constraints. The introduction of new constraints broadens the domain and this allows an easier resolution of the problem.  
  
In the combined RC1 model, we added Channeling Constraints, which plays an important role: it allows us to maintain consistency between the variables in the two models. The combined model has several benefits, including improved propagation.  
Thanks to Channeling Constraints, we are able to use constraints to maintain consistency between the variables in the two models: these additional constraints help identify good solutions more efficiently, since the constraints help "propagate" information more quickly between the variables involved.  
  
We explained ourselves wrongly. Removing the constraints does not affect correctness because the constraints are implicitly present in the others. In fact, as we wrote for the sequence puzzle, the implicit constraints only give extra information and allow more propagation for the model: this allows us to arrive at the solution more quickly. Removing the constraints does not lead to erroneous results but only to a small drop in performance.

**- R -> RC1**

"Broadining the domain" is not concept in CP, so I don't know what you mean by that, better avoid it

We don't have an optimization problem here, so all solutions are equally good, hence no concept of "good solutions" either

The answer to the question is indeed here: " .. since the [channelling] constraints help propagate information more quickly between the variables involved." The domain reductions in one model propagates to the second model (via the channelling constraints), which can invoke further reductions in that model which can then be propagated to the initial model (again via the channelling constraints) and so on...This is the main reason why a combined model bring about more constraint propagation.

**- R1 -> R2 -> RC3**

yes, indeed.

Well done!