



DeepL

DeepL Proに登録すると、より大きな  
詳しくは、[www.DeepL.com/pro](https://www.DeepL.com/pro) を



TECHNISCHE  
UNIVERSITÄT  
DRESDEN

情報科学研究科 情報工学研究所

キャリアチェーンに女王を乗  
せる - No.27 -」。

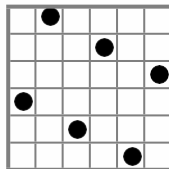
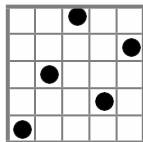
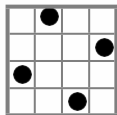


## 旅程表

- 問題点と複雑さの概要
- ソリューションアプローチ
- ハードウェアのマッピングと最適化
- $N=27$ の場合の継続的な計算

# $N$ -Queens Puzzle

- $N \times N$  のチェスボードに 攻撃力のないクイーンを  $N$  個配置する :



- 汎用的なソリューションテンプレート？
- 何（根本）解決？

# モチベーション

$N$ -Queens Puzzleの探検は：

- an embarrassingly parallel、
- を容易に拡張することができます、
- 計算量に制限のある

ワークロード

として機能します：

- 窮屈なデザインに取り組むための訓練用オブジェクトです：
  - 効率的なコーディングとリソース利用を実現します、
  - ツーリングとパラメータ探索を行います。
- ツールやデバイスのベンチマークになります。

そして、そうです：私たちはただできるのです！

## 既知の溶液の数

$N$	ソリュー ション	$N$	ソリューション
1	1	14	365596
2	0	15	2279184
3	0	16	14772512
4	2	17	95815104
5	10	18	666090624
6	4	19	4968057848
7	40	20	39029188884
8	92	21	314666222712
9	352	22	2691008701644
10	724	23	24233937684440
11	2680	24	227514171973736
12	14200	25	2207893435808352
13	72712	26	22217600616261014

網羅的なバックトラックによる solution 探索は、階乗時間  $O(N!)$  を必要とする。

$N=20$  を超えると非常に難しい。

$N = 25$ :

- フランス INRIA による  
Java グリッド計算。

ランタイムです：

リアル > 6ヶ月  
シミュレーション > 53年

## 既知の溶液の数

$N$	ソリュー ション	$N$	ソリューション
1	1	14	365596
2	0	15	2279184
3	0	16	14772512
4	2	17	95815104
5	10	18	666090624
6	4	19	4968057848
7	40	20	39029188884
8	92	21	314666222712
9	352	22	2691008701644
10	724	23	24233937684440
11	2680	24	227514171973736
12	14200	25	2207893435808352
13	73712	26	22317699616364044

網羅的なバックトラックを行うことで  
潤滑油の探索が必要  
階乗時間  $O(N!)$ 。

$N=20$ を超えると非常に難しい。

$N = 26$ :

- 9ヶ月計算  
をFPGAで完成させる  
2009年7月11日
- 2009年8月30日、ロシアのMC#スーパー  
コンピューティング  
プロジェクトにより  
確認された結果。



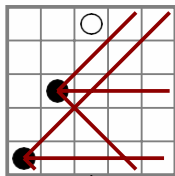


## タックル $N = 26$

- 恥ずかしくなるほどの並列ワークロード：
  1.  $L \ll N$ 列の前置きをします。
  2. サブボードを**単独**で探索する。
  3. 小計を収集し、集計する。
- 分散コンピューティングに最適です：
  - インターネット (BOINC) → NQueens@Home
  - FPGAです！ → クイーンズ@TUD
    - インテリジェントなFPGA実装により、世界規模の分散計算の威力に挑戦する。
    - 2008年11月7日にオーバーフローのバグを特定し、報告しました。
    - これにより、 $N = 24$ の解に関する未解決の論争が解決されました。  
自前で計算することなく

## アルゴリズムの概要

排他的バックトラックによる解の探索。



はい

1. マークが探りを入れた。
2. ブロックベクタを更新する。
3. 次のコラムに進む / 解決策をカウントする。

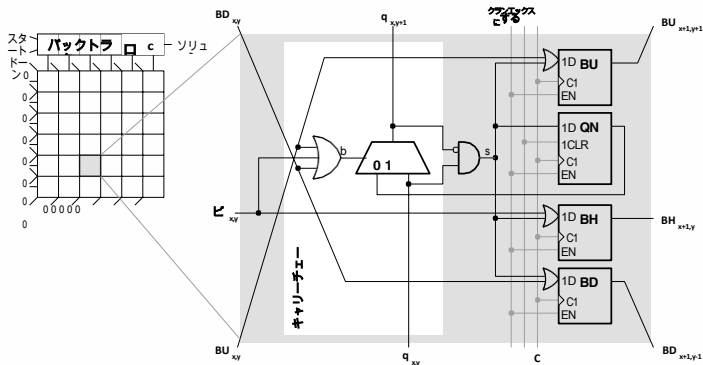
いいえ

1. マークがはっきりしている。
2. 前のコラムに戻る / 終了しました。

有効な配置をまだ探っていない  
のでしょうか？

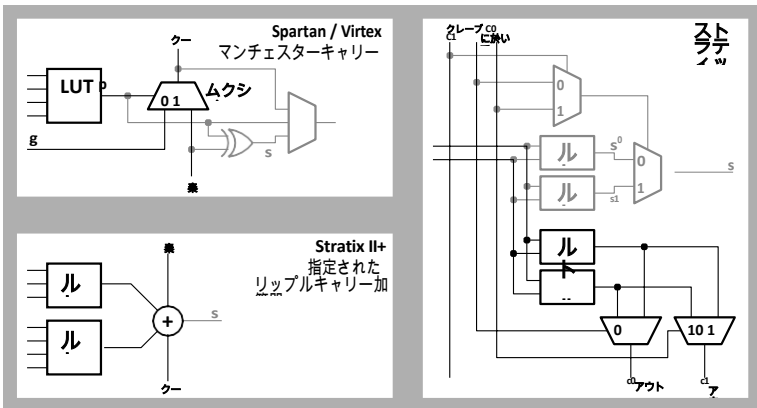
*Blocking Vectors* を計算することで、頻繁な制約の検証を回避することができます。

# FPGAマッピング



キャリーチェーンを使用して、1つの列を1つの高速クロックサイクルで処理します。

## キャリーチェーンストラクチャー



2進数のワード加算を高速化するためにキャリーチェーンを実装しています。

## 加算による汎用キャリアチェーンマッピング

### 1. 派生キャリア／トークン伝搬

ケース	サイズ プラス1	商品説明
$k_i$ : キル	0	鏹一文
$p_i$ : 伝播 (でんぱ) する	$c_i$	パス・ア・キャリア: ノー・クイーンのままブロックされる。
$g_i$ : ジェネレート	1	常にキャリア: 現在のクイーン配置を保持

### 2. 加算の決定

$$a_i = g_i + p_i b_i = g_i$$

### 3. Sum $s \leq a + b$ からトークンを推論する。

入ってくるキャリア／トークンの使用に依存する方程式で：

$$c_i = s_i \oplus p_i$$

ザイリンクス社製デバイスへのマッピングは、最適化された実装を使用しています。

TU Dresden

キャリアチェーンに女王を乗せる- No.27 -。

21枚中10枚目

## プッシュ型パフォーマンス

小型化と高クロック化に向けた最適化：

- アクティブカラムを1つ維持する。
- シフトされたレジスタのプレーンアレイ内に配置された列を保持します。
- すべての行と対角線に、それぞれ配置と後退の設定と解除を行うグローバルブロックベクトルを使用します。

なお、これはソフトウェアではかなり高価なものです！

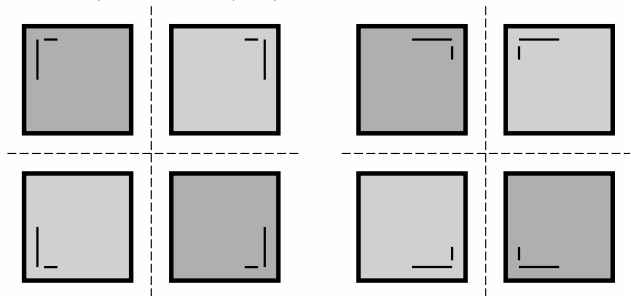
## デザイン $N = 27$

列ベースの事前配置は、線対称性を利用して探索空間を半分にすることができる。

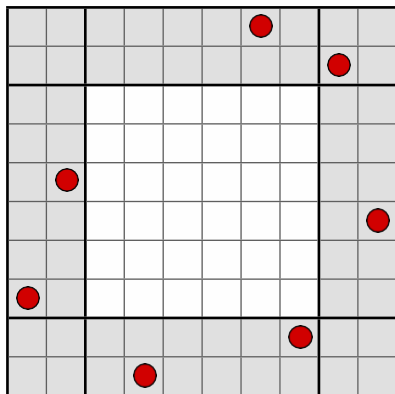


## デザイン $N = 27$

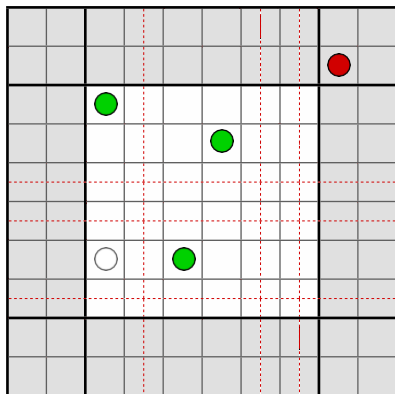
列ベースのプリプレースメントでは、線の対称性を利用して検索スペースを半分にする事ができる。まだまだありますよ：



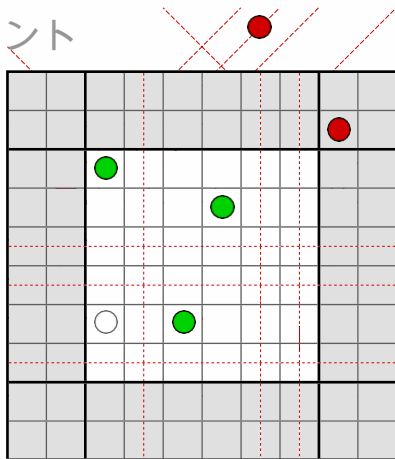
## コロナプリプレイスメント



## コロナプリプレイスメント



# コロナプリプレースメント



アドバンテージがあります：

- 検索スペースを8分の1に削減。

課題（解決済み）です：

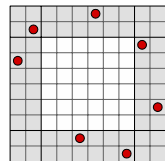
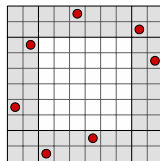
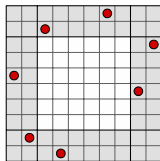
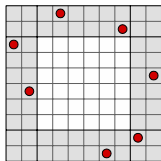
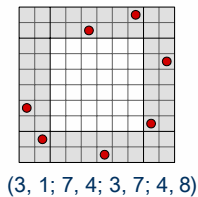
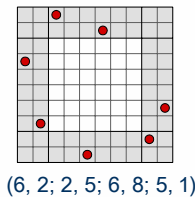
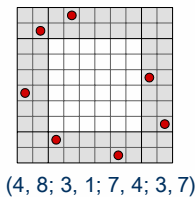
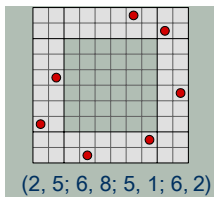
- *canonical representative*を定義する。
- の解を数える。  
自己対称的なプリプレースメント  
を正しく行うことができます。

2.024.110.796コロナルプリプレースメントを $N=27$ で実施しました。

1秒間に1つ解くと、64年分の計算時間が必要です。

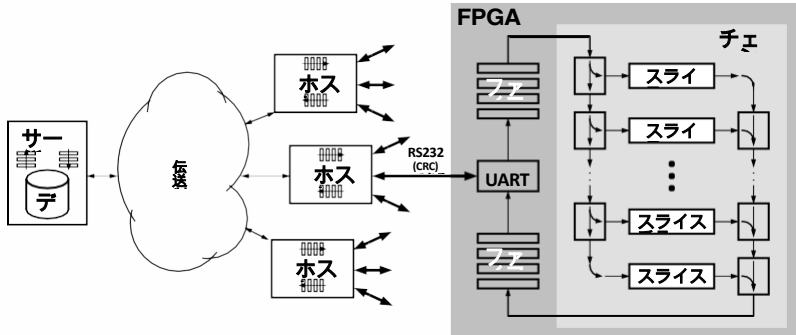


## プリプラです：キヤノンの代表



(7, 4; 3, 7; 4, 8; 3, 1) (5, 1; 6, 2; 2, 5; 6, 8) (3, 7; 4, 8; 3, 1; 7, 4) (6, 8; 5, 1; 6, 2; 2, 5)  
 形質の辞書の順序で決定される最小値。

# プロジェクトインフラ



## スケーラビリティ

本質的に計算量に制限がある：サブ問題の解は21バイトでエンコードされるだけである。現在のピークは1秒間に25個の解、すなわち4.2kBit/sの正味ペイロードです。

プロトコルのオーバーヘッドを100%と仮定すると、サーバー側で成熟した100MBit/sのインターフェースを使い切った場合、2.5時間で完全に終了することを意味する。





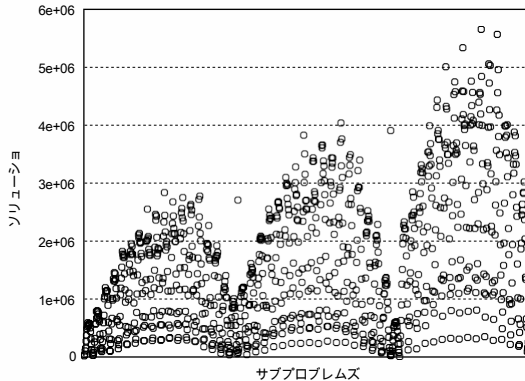
## 貢献するデバイス

ボード	デバイス	ソルバ ー	時計	エ ス イ ー
VC707	XC7VX485T-2	325	250.0 MHz	812
KC705	XC7K325T-2	241	290.4MHz	700
ML605	XC6VLX240T-1	125	200.0MHz	250
ディーイーフォ ー	EP4SGX230KF40C2	125	250.0 MHz	312
DNK7_F5_PCl_e	5×XC7K325T-1	5× 240	220.0MHz	2640

SE ( Solver Equivalent ) : 100MHzで動作するソルバーユニット1台分



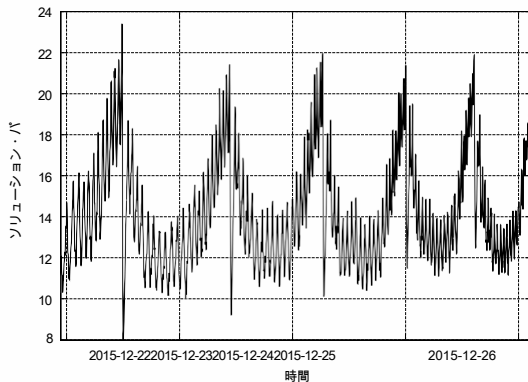
# エマージェント・パターンズソリューションカウ ントス



# (辞書的に並べられた最初の1000個の部分問題)

キャリアチェーンに女王を乗せる-No.27-。

# コンピューテーショナル・スナップショット



## 情勢

現在のところ：

- 中断のない動作において、1秒間に平均15回の解答を達成し
- 2,024,110,796個の部分問題のうち2.7%が解けた。

継続的な取り組み

- ローカルクロック資源（*BUFR*、*BUFH*）の利用を検討し、より高い性能を引き出す。
- GPUへの移植は現在開発中です。

# ありがとうございます！

実装はすべてオープンソースとして公開されています：

*<https://github.com/preusser/q27>*