**Problema:** N valdovių išdėstymas NxN šachmatų lentoje taip, kad nekirstų viena kitos.

**Rezultatas:** kiek yra variantų išdėstyti N valdovių NxN lentoje.

**Darbo priemonės:** MPI kalbos galimybės, universiteto super kompiuteris.

**Sprendimas:** kiekviena pirmo stulpelio eilutė deda valdovę į savo eilutę ir pereina į antrą stulpelį, kuriame tiria kiekvieną langelį, ar jame galima saugiai pastatyti valdovę:

* Jeigu galima, tai pereinama į kito stulpelio analizę
* Jeigu negalima, tai pereinama prie sekančios eilutės tikrinimo, ar saugu joje pastatyti valdovę.

Taigi nėra tikslo skirti daugiau procesorių negu nurodomas N.

Realizuota valdovo-vergo principu.

Valdytojas (master‘is)

Vergas (slave)

Dalina darbus (MPI\_RECV)

Prašo darbo (MPI\_SEND)

Ar yra darbų?

Ar gavo darbą?

Taip

Taip

Ne

Ne

Duoda darbą (MPI\_SEND)

Siunčiami pranešimai apie pabaigą (MPI\_SEND)

Baigia darbą

Skaičiuoja

**Spartėjimas:** spartėjimas yra beveik lygus branduolių skaičiui.

**Išvados:** Programa dirbanti pagal šį algoritmą spartinasi labai gerai, beveik tiesinis spartėjimas. Atlikus tyrimus su JAVA ir MPI programavimo kalbomis pastebėtas skirtumas, jog tas pats algoritmas su JAVA programavimo kalba dirba greičiau laiko atžvilgiu, nors spartėjimo ar plečiamumo atžvilgiu didelio skirtumo nepastebėta.