Investigation

For "rye-s-93.crs" file:

* Total Courses: 486

1. Strategy: Slots=22, Shift=22, Iters=1000
   * Result:
     + Shift=22, Minimum Clashes = 102 at Step 1
2. Strategy: Slots=17, Shift=17, Iters=1000
   * Result:
     + Shift=17, Minimum Clashes = 642 at Step 1
3. Strategy: Slots=20, Shift=5, Iters=1000
   * Result:
     + Shift=5, Minimum Clashes = 222 at Step 10
4. Strategy: Slots=70, Shift=80, Iters=1000
   * Result:
     + Shift=80, Minimum Clashes = 0 at Step 1
5. Strategy: Slots=2, Shift=1, Iters=10000
   * Result:
     + Shift=1, Minimum Clashes = 4600 at Step 25
6. Strategy: Slots=30, Shift=30, Iters=1000
   * Result:
     + Shift=30, Minimum Clashes = 0 at Step 1

For "yor-f-83.crs" file:

* Total Courses: 181

1. Strategy: Slots=17, Shift=33, Iters=1
   * Result:
     + Shift=33, Minimum Clashes = 770 at Step 1
2. Strategy: Slots=50, Shift=50, Iters=1000
   * Result:
     + Shift=50, Minimum Clashes = 0 at Step 1
3. Strategy: Slots=300, Shift=250, Iters=1000
   * Result:
     + Shift=250, Minimum Clashes = 0 at Step 1
4. Strategy: Slots=30, Shift=250, Iters=100
   * Result:
     + Shift=250, Minimum Clashes = 0 at Step 1
5. Strategy: Slots=30, Shift=23, Iters=1000
   * Result:
     + Shift=23, Minimum Clashes = 0 at Step 2
6. Strategy: Slots=2, Shift=2, Iters=100
   * Result:
     + Shift=2, Minimum Clashes = 8414 at Step 1
7. Strategy: Slots=40, Shift=50, Iters=1000
   * Result:
     + Shift=50, Minimum Clashes = 0 at Step 1

Slots:

* Increase in Slots:
  + Generally led to a reduction in clashes due to increased flexibility in scheduling courses.
  + However, excessively high slot numbers might lead to increased computational complexity without significant clash reduction.
* Decrease in Slots:
  + Initially might reduce clashes by forcing courses into fewer available timeslots.
  + However, too few slots could result in increased clashes due to limited scheduling options.

Shifts:

* Increase in Shifts:
  + Often correlated with a decrease in clashes as it provides more opportunities to reschedule courses.
  + However, extremely high shifts might not necessarily yield further clash reduction and could lead to computational inefficiency.
* Decrease in Shifts:
  + May lead to an initial decrease in clashes due to courses being scheduled more tightly.
  + However, overly restrictive shifts might exacerbate clashes by limiting scheduling flexibility.

Iterations:

* Increase in Iterations:
  + Typically allows the algorithm more time to refine the timetable, potentially leading to further clash reduction.
  + However, diminishing returns might be observed beyond a certain point, where additional iterations yield minimal improvements.
* Decrease in Iterations:
  + Might lead to quicker results but could sacrifice the quality of the final timetable, especially if clashes persist.

Conclusion:

* The investigation results demonstrate the intricate balance between slots, shifts, and iterations in minimizing clashes within timetables.
* Optimal solutions often lie in a nuanced combination of these parameters, tailored to the specific characteristics of the input dataset.
* Further fine-tuning and experimentation are necessary to achieve the most efficient and effective scheduling outcomes.
* Continuous refinement of the algorithm based on these observations could lead to significant improvements in timetable optimization processes.