Research Review

TASKS:

- After completing the coding and analysis portion of the project, read up on **important historical developments** in the field of AI planning and search. Write a one-page report on **three** of these developments, highlighting the relationships between the **developments** and their **impact** on the field of AI as a whole.
- Appropriate sources (such as books or magazine or journal articles) should be cited, and you should use citations in-line for sourced facts, quotations, and inferences.

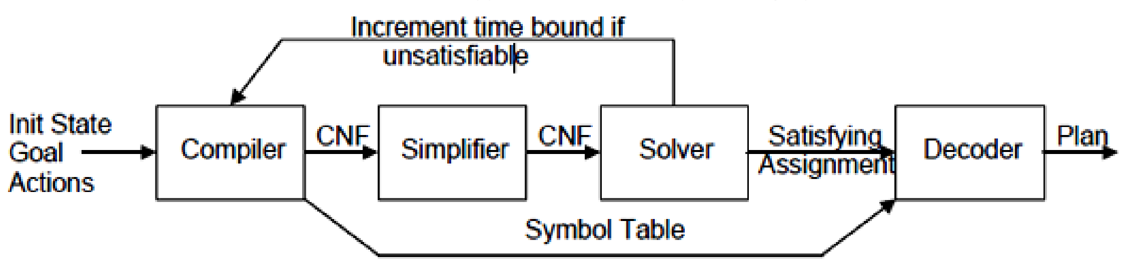
Important Historical Developments in the field of AI Planning and Search

[1] Compilation of Planning problems into SAT

Even though the early formulation of planning in the Theorem Proving space, most researchers have long assumed that special purpose planning algorithms (TWEAK, SNLP, UCPOP and GraphPlan) were a necessity for practical performance. However, recent improvements in the realm of propositional satisfaction methods are calling this assumed requirement into doubt. Recent experiments suggest that compiling planning problems into SAT (Boolean Satisfiability Problem) might yield the world's fastest STRIPS-style planner.

The compiler takes a planning problem as input, guesses a plan length, and generates a propositional logic formula, which if satisfied, implies the existence of a solution plan; a symbol table records the correspondence between propositional variables and the planning instance. The simplifier uses fast (linear time) techniques such as *unit clause propagation* and *pure literal elimination* to shrink the CNF formula. The solver uses systematic or stochastic methods to find a satisfying assignment which the decoder translates (using the symbol table) into a solution plan. If the solver finds that the formula is unsatisfiable, then the compiler generates a new encoding reaching a longer plan length.

Architecture of a typical SAT-based planning system



[2] Hybrid Planning

Research and experiments have shown that that there is *no single planning strategy suited to all planning problems*. All of the approaches discussed in [2] this paper display a wide range of performance over different problem types the structure and organization of the problem determine the extent to which a particular strategy is suited to a particular problem. For this reason hybrid planning approaches can be very successful. However, determining *automatically* which of the available strategies in a hybrid to invoke remains a *difficult* research problem. In the way of examples, a Graphplan strategy with a model-checking approach, using time bounds to determine which of the strategies to apply to a problem. The Graphplan strategy was tried first, and if no solution was found within the time bound the problem would be reformulated as a model-checking problem.

It is often the case that these specialized problems appear as sub-problems within a larger planning problem. As an example, a problem that involves transporting components between locations, constructing complex artifacts with the components and then delivering them to customers can be seen to contain (route planning, resource allocation, job-shop scheduling and construction planning) sub-problems. Each of these can, to some extent (although *not* entirely) be decoupled from the others and solved using specialized technology. Several other examples have now been demonstrated in which a successful automatic identification of sub-problems is combined with automatic configuration of specialized problem-solving technology and solution of planning problems using these specialized solvers

[3] Parallelism

An *operator application* in general means that the *effects* become true and "something else", unaffected **fluents** preserve their truth values (as opposed to "affected" fluents). However, it is **not** necessary to *restrict to one operator application at a time*: several operators can often be applied in **parallel**. This *could* reduce the planning effort significantly because *separately considering* **n!** different (albeit behaviorally equivalent) orderings of **n** mutually independent operators is avoided.

The parallel application of operators is well-defined *when the operators do **not** interact*. This means that the result of applying the operators in **any order** is possible and has the **same effect**. A sufficient *condition* for this is that (1) the *operators do not have contradictory effects,* and (2) none of the operators falsifies the preconditions of any other. Allowing parallel application still more liberally is possible, and this sometimes leads to even more efficient planning.

References:

[1] Recent Advances in AI Planning:
https://homes.cs.washington.edu/~weld/papers/pi2.pdf

[2] Progress in AI Planning Research and Applications:
https://www.researchgate.net/publication/242415929_Progress_in_AI_Planning_Research_and_Applications

[3] An overview of recent algorithms for AI Planning:
http://www.cs.toronto.edu/~sheila/2542/w06/readings/RintanenHoffmann01.pdf