There are a number of planners available to use with Moveit!. These include :
1) OMPL - an open-source motion planning library
2) STOMP -Stochastic Trajectory Optimization for Motion Planning
3) CHOMP -Covariant Hamiltonian optimization for motion planning
4) SBPL - Search Based Planning library.
As it can be see from the official ROS wiki page, currently OMPL is the
only planner fully supported leaving the others to be somewhere between partially supported to work in progress.

There are both analytical and numerical solutions for Inverse Kinematics. Analytical solutions suffer from an inability to generalize to tool-use scenarios or changes in robot configuration, as the solver must be constructed beforehand. Typically, Numerical IK solvers are more generic in that they rely on a frequent,runtime approximation of the local inverse Jacobian in order to try to find joint solutions that come “close enough” to the desired Cartesian solution. Numerical IK methods use the Newton method or similar to iterate until the solution is found. While theoretically sound, numerical approaches can be quite slow compared to analytical approaches, and thus there is active research to try and speed up the computation of the Jacobian, speed up the matrix inversion, and converge to a quality solution without getting continuously stuck in bad local minima.

The IK solvers that I encountered include track IK,KDL solver. the KDL solver is ideally suited for kinematic chains having a DOF greater than 6 and have a restriction where a joint cannot have more than 1 DOF. track_IK solver runs two different IK methods and combines the result to outperform the popular KDL solver. Also trac_IK handles joint-limited chains better than KDL without increasing solve time. However one should also note that significant improvements over KDL solver only surface when for large chains which is not the case with cool400 robot.

For a single IK solver request,TRAC-IK spawns two solvers, one running SQP-SSand one running KDL-RR. Once either finishes with a solution, both threads are immediately stopped, and the resulting solution is returned.

I have configured KDL solver and Track ik for cool400 robot and implimented motion planning with the help of OMPL.

Type of Inverse Kinematic solver	Computation time(s)	Path generated
KDL_Solver	0.046	Path 1
Track_IK solver	0.020	Path 2

Unable to reach any conclusion regarding computation time since the paths are different. Large number of data set regarding computation time for different paths are required to know which solver works best in terms of computation time and efficiancy.