Abstract
The CMS tracker is the largest silicon detector ever built, covering 200 square meters and providing an average of 14 high precision measurements per track. Tracking is essential for the reconstruction of objects like jets, muons, electrons and tau leptons starting from the raw data from the silicon pixel and strip detectors. Tracking efficiency is widely used also at trigger level as it improves objects tagging and recognition. The CMS tracking code is organized in several levels, known as ‘iterative steps’, each optimized to reconstruct a class of particles, trajectories, as the ones of particles originating from the primary vertex or displaced tracks from particles resulting from secondary vertices. Each iterative step consists of seeding, pattern recognition and fitting by a Kalman filter, and a final filtering and cleaning. Each subsequent step works on hits not yet associated or reconstructed particle tracks. This approach is expected to provide an important improvement of efficiency and reduction of fake hits, thus making CMS track reconstruction more robust against the pileup expected in 2015. The CMS tracking code is continuously evolving to make the reconstruction computing load compatible with the increasing instantaneous luminosity of LHC, resulting in a large number of primary vertices and tracks per bunch crossing. This is achieved by optimizing the iterative steps and by developing new software techniques. Tracking algorithms used in CMS are described; physics and computing performances are discussed with respect to Run 1 and Run 2 physics program and within CMS future upgrades.

The CMS tracker, the world largest Silicon detector
Pixel - 64m m channels, 100x150 µm pixel x pixel Strip: 15148 Silicon strip modules, pitch in the range 80 to 200µm, ~200m² of Silicon active area, ~900 channels with full optical analog readout.

CMS tracking procedure
Seeding proto-tracks are searched for in the innermost layers (pairs + primary vertex, triplets).

Tracking
Tracks are produced for the subset of tracks passing the trigger; specific iterations can be introduced for special physics requirements.

CMS tracking parameters in 2012
The CMS efficient tracking relies on iterations (steps) of the tracking procedure; each step works on the remaining not-yet-associated hits. Each step is optimized with respect to its seeding topology, and to the final quality cuts. The iterative tracking is largely configurable: it can be adapted to be used within High Level trigger. Specific iterations can be introduced for special physics requirements. Each step is optimized to reconstruct specific particles or jets, with a high-purity selection of seeds or high-purity seed selection.

Iterative tracking
The CMS efficient tracking relies on iterations (steps) of the tracking procedure; each step works on the remaining not-yet-associated hits. Each step is optimized with respect to its seeding topology, and to the final quality cuts. The iterative tracking is largely configurable. It can be adapted to be used within High Level trigger. Specific iterations can be introduced for special physics requirements. Each step is optimized to reconstruct specific particles or jets, with a high-purity selection of seeds or high-purity seed selection.

Efficiency and fake rate
With pileup and without pileup, the tracking efficiency results are for charged particles produced less than 3 cm from the centre of the beam spot in (30 cm) from the centre of the beam spot in.

Efficiency
The CMS tracking will need to be adapted for the High Luminosity Phase of LHC from 2014 on. Complete software and calibration optimizations will be required for the high luminosity operation. The tracking is tuned to give the best performances within 2015. The tracking efficiency that increases with the instantaneous luminosity. This yields to a less impact on tracking with large PU since the pixel detector suffers a dynamic inefficiency that increases with the instantaneous luminosity. This yields to a less effective pixel based seeding that results in an enlarged load on strip based iterative step having an intrinsically less accurate (large search windows in a building) and more prone to combinatorics. This is clearly visible in the tracking timing plots at different PU. The effect of pileup is dramatic on iterations seeded by pairs of strip matched hits (iter (0-2)) and iter (4-5), while for steps seeded by pixel pairs (iter (2)) and iter (5-6). Pixel triplet seeded steps (iter (6-7)) show a behavior with respect to PU. This analysis clearly shows where to act to make CMS tracking more robust for high luminosity.

Outlook for Run2
The CMS tracking needs to be continuously adapted for increased instantaneous luminosity of LHC. Run2 starting in 2015 makes no exception. By definition the tracking is tuned to give the best performances within the limited computing resources allotted for reconstruction. Track reconstruction is the largest chunk of CMS reconstruction with respect to the computing resources. In 2012 maximum pile-up was ~20 events per bunch crossing. In 2015 up to 45 is expected. Moreover LHC will change bunch crossing frequency from 50 to 25 ns yielding to hits due to out-of-time pile-up (~5% in pixel detector and ~45% in silicon detector). The combinatorial problem behind the track building is largely increased. The concept of overlapping (i.e. progressive removal of hits attached to tracks) is less evident for momentum and less granular detectors.

Tracking efficiency and fake rate
For each of the last three iterations, a global Kalman filter step is applied to all tracks to remove false hits. The efficiency that increases with the instantaneous luminosity. This yields to a less effective pixel based seeding that results in an enlarged load on strip based iterative step having an intrinsically less accurate (large search windows in a building) and more prone to combinatorics. This is clearly visible in the tracking timing plots at different PU. The effect of pileup is dramatic on iterations seeded by pairs of strip matched hits (iter (0-2)) and iter (4-5), while for steps seeded by pixel pairs (iter (2)) and iter (5-6). Pixel triplet seeded steps (iter (6-7)) show a behavior with respect to PU. This analysis clearly shows where to act to make CMS tracking more robust for high luminosity.

Tracking efficiency and fake rate vs. |η| and |ϕ| for Run1 and Run2
Tracking efficiency and fake rate vs. |η| and |ϕ| for Run1 and Run2. For Run1, the tracking is undergoing an extended optimization based on several cornerstones. The extended usage of Triplet Base Seeding
Development of a new triplet algo to replace iter #5 and iter #6 pair strip-seeded; the new algo is based on a χ² cut from straight line fit of 3 points in the η plane and also from a higher binomial constraints; half of the seeds are generated but the same number of tracks are reconstituted. At PU=40, the total time reduction on tracking is ~40% with negligible efficiency, thanks to an timing improvement on η and t₀_aline less than a factor 2.

Strip Cluster Charge cut
With 25 ns bins, the increase in occupancy due to out-of-time PU for the strip detector induce an increase by 2x both on tracking and time rate; however clusters from out-of-time PU are characterized by low collected charge since it is induced by tracks (loops) that deposit charge not in time with respect to the sampling windows. Cutting on the cluster charge largely supresses the hits due to out-of-time PU. A cut on cluster charge to be applied upfront, during seeding or during pattern recognition is foreseen for Run3. The cut takes into account the occupancy and the complexity of the alignment on an event by event basis and is designed to be |P| dependent to optimise the potential signal from fractional charge particles (baryons) with some BSM models. On the detector side, it requires stable performance to be ensured by introducing tracker gain calibration in the automatic calibration procedure (Prompt Calibration). This is achieved by embedding a rate dependent gain calibration in the Track Trigger Global (TTG) calibration loop.

Other Tracking Optimization
Faster first: Since iter (3) is faster (per track) than iter (1) and (2). By moving iter (3) right after iter (0). Iter (4) was simplified by removing redundant seed combinations, but lost tracks are recovered in iter (5). Optimization of track selection Less iteration-specific track selection removes overlaps between iterations and improves overall performances. Code optimizations: Better use of magnetic field, improved architecture and thread safety for event parallelization. These improvements allow for a 10% gain in timing with no performance loss.

A glimpse into the future
CMS tracking will need to be adapted for the High Luminosity Phase of LHC from 2014 on. Complete software rewrite will require new framework and algorithm vectorization and parallelization and innovative tracking algorithms.