```
Section Files
Map                     ; Map of the environment we'll use to navigate


Section Localization settings
NumSamples 2000         ; minumum 0,  No of pose samples for MCL. The larger
                        ; this number, the more computation will localization
                        ; take. Too low a number will cause the robot to lose
                        ; localization. This is also the maximum no of samples
                        ; which will be used for localization if no of samples
                        ; are varied along with the localization score.

GridRes 100             ; minimum 10,  The resolution of the occupancy grid
                        ; representing the map in mm. Smaller resolution
                        ; results in more accuracy but more computation.

PassThreshold 0.2       ; range [0, 1],  After MCL sensor correction, the
                        ; sample with the maximum probablity will have a score
                        ; based on the match between sensor and the map
                        ; points. This is the minimum score out of 1.0 to be
                        ; considered localized.

KMmPerMm 0.05           ; minimum 0,  When the robot moves linearly, the error
                        ; in distance is proportional to the distance moved.
                        ; This error is is given as a fraction in mm per mm

KDegPerDeg 0.05         ; minimum 0,  When the robot rotates, the error in the
                        ; angle is proportional to the angle turned. This is
                        ; expressed as a fraction in degs per deg.

KDegPerMm 0.0025        ; minimum 0,  When the robot moves linearly it can
                        ; also affect its orientation. This drift can be
                        ; expressed as a fraction in degs per mm.

TriggerDistance 200     ; minimum 0,  Since MCL localization is
                        ; computationally expensive, it is triggered only when
                        ; the robot has moved this far in mm.

TriggerAngle 5          ; minimum 0,  Since MCL localization is
                        ; computationally expensive, it is triggered only when
                        ; the robot has rotated this far in degs.

TriggerTimeEnabled false ; This flag will decide if the localization will be
                        ; called every 'TriggerTime' msecs. Once this flag is
                        ; true the IdleTimeTrigger* parameters will take
                        ; effect. This feature is meant to take care of cases
                        ; when the robot has not moved much for a time and the
                        ; position should be refined .

TriggerTime 10000       ; minimum 1500,  Once the TriggerTimeFlag is set to
                        ; true this parameter will decide how long the robot
                        ; has been idle in milli seconds before it starts a
                        ; localization near the last known robot pose.

IdleTimeTriggerX 200    ; minimum 0,  When localization is triggered by idle
                        ; time this parameter decides the range of the samples
                        ; in X coords in mm.

IdleTimeTriggerY 200    ; minimum 0,  When localization is triggered by idle
                        ; time this parameter decides the range of the samples
                        ; in Y coords in mm.

IdleTimeTriggerTh 15    ; minimum 0,  When localization is triggered by idle
                        ; time this parameter decides the range of the samples
                        ; in Theta coords in degs.

RecoverOnFail false     ; If localization fails, this flag will decide if a
                        ; static localization is attempted around last known
                        ; robot pose. Such a reinitialization can cause the
                        ; robot to be hopelessly lost if the actual robot is
                        ; very different from its known pose

FailedX 300             ; minimum 0,  Range of the box in the X axis in mm to
                        ; distribute samples after localization fails.

FailedY 300             ; minimum 0,  Range of the box in the Y axis in mm to
                        ; distribute samples after localization fails.

FailedTh 45             ; minimum 0,  Range of the angle in degs to distribute
                        ; samples after localization fails.

PeturbX 10              ; minimum 0,  After sensor correction and resampling
                        ; the chosen pose is perturbed to generate a new
                        ; sample. This parameter decides the range to peturb
                        ; the X axis in mm.

PeturbY 10              ; minimum 0,  After sensor correction and resampling
                        ; the chosen pose is perturbed to generate a new
                        ; sample. This parameter decides the range to peturb
                        ; the Y axis in mm.

PeturbTh 1              ; minimum 0,  After sensor correction and resampling
                        ; the chosen pose is perturbed to generate a new
                        ; sample. This parameter decides the range to peturb
                        ; the angle in degs.

PeakStdX 10             ; minimum 0,  Extent of the ellipse in the X axis in
                        ; mm beyond which the sample poses will be considered
                        ; multiple localizations after resampling.

PeakStdY 10             ; minimum 0,  Extent of the ellipse in the X axis in
```

```
                         ; mm beyond which the sample poses will be considered
                         ; multiple localizations after resampling.

PeakStdTh 1              ; minimum 0,  Extent of the angle in degs beyond which
                         ; the sample poses will be considered multiple
                         ; localizations after resampling.

PeakFactor 1e-06         ; range [0, 1],  When a no of samples have non zero
                         ; probabilities such as when there is ambiguities in a
                         ; corridor. This is the threshold below the maximum
                         ; probablity to be considered a valid hypothesis.

StdX 400                 ; minimum 0,  The standard deviation of the gaussian
                         ; ellipse in X axis in mm at start of localization.

StdY 400                 ; minimum 0,  The standard deviation of the gaussian
                         ; ellipse in Y axis in mm at start of localization.

StdTh 30                 ; minimum 0,  The standard deviation of the gaussian
                         ; angle in degs at start of localization.

SensorBelief 0.9         ; range [0, 1],  Probablility that a range reading
                         ; from the laser is valid. This is used in the
                         ; correction of the probablities of the samples using
                         ; the sensor.

OccThreshold 0.1         ; range [0, 1],  The threshold value of the occupancy
                         ; grid to consider as occupied.

AngleIncrement 0         ; range [0, 180],  Only the laser readings which are
                         ; this many degrees apart are used for the
                         ; localization. The lower limit is decided by the
                         ; LaserIncrement setting

DiscardThreshold 0.33    ; range [0.33, 1],  A robot sample pose lying inside
                         ; an occupancy grid cell with a value above this will
                         ; be usually discarded Useful in cases where robot may
                         ; intersect map points such as during patrolbot
                         ; docking

FuseAllSensors false     ; ARNL uses a Kalman filter which allows you to
                         ; combine the data from the MCL localization, the
                         ; movement from the encoder between cycles and
                         ; reflectors if mapped and seen by the laser. This
                         ; advanced feature can be disabled to revert to the
                         ; basic MCL localization, using this flag

ReflectorVariance 10000  ; minimum 0,  This number will be used as the variance
                         ; of the (x, y) coords of the center of the reflectors
                         ; in the R matrix of the Kalman filter.

Qxx 100                  ; minimum 0,  This is the first element of the
                         ; diagonal covariance matrix which will define the
                         ; error in the kalman model for the X axis.

Qyy 100                  ; minimum 0,  This is the second element of the
                         ; diagonal covariance matrix which will define the
                         ; error in the kalman model for the Y axis.

Qtt 1                    ; minimum 0,  This is the third element of the
                         ; diagonal covariance matrix which will define the
                         ; error in the kalman model for the Theta axis.

ReflectorMatchDist 1000  ; minimum 0,  When finding the closest reflector in
                         ; the map to an observed reflection, this is the
                         ; maximum distance the system will search to find the
                         ; closest reflector.

ReflectorMaxRange 32000  ; minimum 0,  This is the maximum distance that the
                         ; SICK lrf is capable of seeing a reflector. (This is
                         ; smaller than the max range of the regular SICK
                         ; readings)

ReflectorMaxAngle 45     ; minimum 0,  This is the maximum angle of incidence
                         ; that the SICK lrf is capable of seeing a reflector
                         ; at. (This is much smaller than the angle that the
                         ; regular SICK readings are capable of returning)

UseReflectorCenters true ; The Kalman filter matches the returns from the laser
                         ; reflectors to the reflectors in the map. When there
                         ; are more than one ray from one reflector, this flag
                         ; will bunch the rays into groups and match the center
                         ; ray with the center of the reflector. If the flag is
                         ; set to false, the rays are matched to the point on
                         ; the reflector by line intersection. This involves
                         ; more computation but may be more accurate.

Triangulate false        ; The regular Kalman filter tries to fuse the encoder
                         ; pose and the data from the reflectors. This is an
                         ; incremental process which will eventually converge
                         ; to the right pose within a few iterations depending
                         ; on the uncetainity models of the sensors. But, when
                         ; the robot is lost, and it sees more than one
                         ; reflector, the pose can be computed directly using a
                         ; closed form solution. The key qualifier is that the
                         ; curent pose of the robot is close enough to match
                         ; the reflectors.

TriangulateScoreThreshold 0.5 ; When the map of the environment is very close
                         ; to the actual environment the robot encounters, the
                         ; triangulation using a few reflectors can actually
```

```
                         ; worsen the localization. This is due to the fact
                         ; that the uncertainity in combining the few
                         ; reflectors and their location usually is worse than
                         ; the uncertainity from using all the laser range
                         ; values from the MCL.The triangulation will kick in
                         ; only if the MCL score drops below this value.

AdjustNumSamplesFlag false ; The number of samples is by default kept high to
                         ; keep the robot from losing localization even after
                         ; initialization. This number can be lowered during
                         ; motion in places of the map where the localization
                         ; score is high to reduce the computation load. Set
                         ; this flag to true if you want to vary the number of
                         ; samples with the localization score. (As the score
                         ; drops the no of samples will rise)

MinNumSamples 200        ; minumum 0,  When the AdjustSamplesFlag is set to
                         ; true the number of samples is reduced as the
                         ; localization score rises. But, this will be the
                         ; lowest number it will be reduced to.

NumSamplesAngleFactor 1  ; minimum 0,  When the AdjustSamplesFlag is set to
                         ; true the number of samples is reduced as the
                         ; localization score rises. But, when the robot has
                         ; rotated significantly, it needs more samples than if
                         ; it had only moved in translation. A bigger angle
                         ; factor will cause the no of samples to not drop as
                         ; fast when the localization is triggered due to
                         ; rotation.


Section Path planning settings
MaxSpeed 750             ; minimum 0,  Maximum speed during path following in
                         ; mm/sec.

MaxRotSpeed 100          ; minimum 0,  Maximum rotational speed in degs/sec

PlanRes 106.25           ; minimum 0,  The resolution of the grid used for path
                         ; planning in mm. It is best to use an integral
                         ; fraction of the robot half width to allow for more
                         ; free space in tight spaces.

PlanFreeSpace 637.5      ; minimum 0,  Preferred distance from side of robot to
                         ; obstacles. The path planner marks each grid cell
                         ; depending on its distance to the nearest obstacle.
                         ; Any cell within half the robot width of an occupied
                         ; cell will still be non traversable by the robot.The
                         ; cell at half robot width will be traversable but
                         ; will have the maximum cost. The cost of the cells
                         ; beyond half width will progressively decrease. The
                         ; PlanFreeSpace is the distance from the side of the
                         ; robot beyond which the cell cost stops decreasing
                         ; and remains constant. Larger values will cause
                         ; larger excursions around obstacles, (This variable
                         ; is related to the FreeSpacing variable in previous
                         ; versions of ARNL which was measured from the center
                         ; of the robot.)

CollisionRange 2000      ; minimum 0,  The distance from the robot within which
                         ; the obstacles seen by the sensor and those on the
                         ; map are used to compute the local path. (User should
                         ; keep this above (maxlvel*maxlvel)/2/goalDecel to
                         ; allow for smoother motion when encountering unmapped
                         ; obstacles)

UseCollisionRangeForPlanning false ; The robot plans ahead locally for a
                         ; distance based on its speed. This may not be
                         ; sufficient in some cases where the user may want it
                         ; to look ahead till the CollisionRange of the
                         ; sensors. This flag will enable the user to force the
                         ; robot to look at least as far as the distance the
                         ; sensors data is incorporated.

PlanEverytime true       ; Local replanning need to be done only if the current
                         ; path is obstructed. This will decide if local
                         ; planning is done every action cycle anyway? Setting
                         ; it to false will be useful when there are too many
                         ; unmapped obstacles which cause the robot to
                         ; constantly flip flop on its way to goal.

ClearOnFail true         ; If this flag is true, any failure in the local path
                         ; planning will force the cumulative range buffers to
                         ; be cleared

FrontClearance 100       ; minimum 0,  Front clearance in mm of the robot while
                         ; avoiding obstacles.

SlowSpeed 100            ; minimum 0,  Speed below and at which is considered
                         ; slow.

SideClearanceAtSlowSpeed 100 ; minimum 0,  Side clearance in mm of the robot
                         ; while avoiding obstacles when it is moving at or
                         ; below the slow speed.

FrontPaddingAtSlowSpeed 100 ; minimum 0,  Distance in addition to the front
                         ; clearance of the robot while avoiding obstacles when
                         ; it is moving at or below the slow speed. In this
                         ; padding the super max translation deceleration will
                         ; be allowed to engage to keep the obstacle away
                         ; before slamming on eStop.
```

```
FastSpeed 2000           ; minimum 0,  Speed at or above which is considered
                         ; fast.

SideClearanceAtFastSpeed 1000 ; minimum 0,  Side clearance in mm of the robot
                         ; while avoiding obstacles when it is moving at or
                         ; above the fast speed.

FrontPaddingAtFastSpeed 1000 ; minimum 0,  Distance in addition to the front
                         ; clearance of the robot while avoiding obstacles when
                         ; it is moving at or above the fast speed. In this
                         ; padding the super max translation deceleration will
                         ; be allowed to engage to keep the obstacle away
                         ; before slamming on eStop.

SuperMaxTransDecel 1000  ; minimum 0,  The maximum translational deceleration
                         ; allowed for avoiding obstacles. This is used in
                         ; conjunction with the padding parameters to limit the
                         ; wear and tear if emergency deceleration is
                         ; unlimited.

EmergencyMaxTransDecel 0 ; minimum 0,  The emergency translational deceleration
                         ; allowed for avoiding obstacles. This is used in
                         ; conjunction with the padding parameters to limit the
                         ; maximum deceleration allowed even if there is a
                         ; chance for collision. (A zero will make it use the
                         ; computed deceleration or SuperMaxTransDecel however
                         ; large it happens to be)

UseEStop false           ; If this flag is true, any obstacle in the path which
                         ; cannot be avoided by regular robot deceleration will
                         ; cause the software emergency stop to be engaged
                         ; which will result in an almost instantaneous stop.
                         ; (May not be suitable for platforms like the
                         ; wheelchair)

UseLaser true            ; Use the laser for collision avoidance?

UseSonar true            ; Use the sonar for collision avoidance? Due to the
                         ; sonar inaccuracies the obstacles may appear larger.
                         ; This flag can be used to set the sonar off when the
                         ; robot has to navigate through narrow openings such
                         ; as doors.

SecsToFail 8             ; minumum 0,  The time in seconds for which local
                         ; search will continue trying to plan when it fails to
                         ; find a safe local path to move.

LocalPathFailDistance 1000 ; minimum 0,  When the sensors see that the local
                         ; path is blocked, the robot will still be allowed to
                         ; drive this close to the block. This is meant to help
                         ; in cases where the robot senses false obstacles from
                         ; afar and stops too soon such as on ramps. (The
                         ; distance at which the robot actually stops will
                         ; depend on the velocity of the robot)

MarkOldPathFactor 0.75   ; minimum 0,  When the robot operates in an
                         ; environment where there are a number of unmapped
                         ; obstacles, the local path it plans may flip from one
                         ; side of the obstacle to the other between cycles. To
                         ; avoid this, ARNL marks the old path by reducing the
                         ; costs of its cells by this factor. This factor
                         ; should be a value between 0 and 1

GoalDistanceTol 200      ; minimum 0,  Distance in mm to the goal that will be
                         ; considered as reached.

GoalAngleTol 10          ; minimum 0,  Angle in degs to the goal orientation
                         ; that will be as done with orientation.

GoalOccupiedFailDistance 1000 ; minimum 100,  When the sensors see that the
                         ; goal is occupied the robot will still be allowed to
                         ; drive this close to the goal anyway.

GoalSpeed 250            ; minimum 0,  Maximum speed at which end move to goal
                         ; is executed. This value and the robot's inertia will
                         ; decide the goal positioning accuracy. (A value of 0
                         ; will keep the normal driving value)

GoalRotSpeed 33          ; minimum 0,  Maximum rotational velocity at which end
                         ; move to goal is executed. This value and the robot's
                         ; inertia will decide the goal positioning accuracy.
                         ; (A value of 0 will keep the normal driving value)

GoalTransAccel 200       ; minimum 0,  Maximum linear acceleration at which end
                         ; move to goal is executed. (A value of 0 will keep
                         ; the normal driving value)

GoalTransDecel 200       ; minimum 0,  Maximum linear deceleration at which end
                         ; move to goal is executed. (A value of 0 will keep
                         ; the normal driving value)

GoalRotAccel 33          ; minimum 0,  Maximum rotational acceleration at which
                         ; end move to goal is executed. (A value of 0 will
                         ; keep the normal driving value)

GoalRotDecel 33          ; minimum 0,  Maximum rotational deceleration at which
                         ; end move to goal is executed. (A value of 0 will
                         ; keep the normal driving value)

GoalSwitchTime 0.4       ; minimum 0,  Time in secs to switch into end move
                         ; mode in addition to the coasting time. (To allow for
```

```
                        ; slack in the ramping down of the velocity to zero)

GoalUseEncoder true     ; For fine positioning at goal, the robot can switch
                        ; to moving based on its encoder pose only. This flag
                        ; decides this

AlignAngle 10           ; minimum 0,  When the robot is stopped this is the
                        ; minimum angle it will rotate in place to align to
                        ; the planned path before it will move linearly.

AlignSpeed 10           ; minimum 0,  This is the velocity below which the
                        ; robot will do the aligning to path direction before
                        ; linear motion.

HeadingRotSpeed 50      ; minimum 0,  There will be some points on the path
                        ; where the robot will be exclusively rotating to
                        ; reach a heading. The user may wish to set the
                        ; rotational speed in this situation to lower than
                        ; normal especially when dealing with heavier robots
                        ; like the Powerbot. (A value of 0 will keep the
                        ; normal driving value) (This will not override the
                        ; GoalRotSpeed)

HeadingRotAccel 50      ; minimum 0,  There will be some points on the path
                        ; where the robot will be exclusively rotating to
                        ; reach a heading. The user may wish to set the
                        ; rotational accel in this situation to lower than
                        ; normal especially when dealing with heavier robots
                        ; like the Powerbot. (A value of 0 will keep the
                        ; normal driving value) (This will not override the
                        ; GoalRotAccel)

HeadingRotDecel 50      ; minimum 0,  There will be some points on the path
                        ; where the robot will be exclusively rotating to
                        ; reach a heading. The user may wish to set the
                        ; rotational decel in this situation to lower than
                        ; normal especially when dealing with heavier robots
                        ; like the Powerbot. (A value of 0 will keep the
                        ; normal driving value) (This will not override the
                        ; GoalRotDecel)

HeadingWt 0.8           ; range [0, 1],  Heading weight for DWA. Unlike the
                        ; heading objective computed from a destination pose
                        ; on the path, as in the conventional DWA, we use a
                        ; path matching function to match the arc made from
                        ; the velocities with the desired computed path.

DistanceWt 0.1          ; range [0, 1],  Distance weight for DWA. Distance
                        ; refers to the distance to collision if any if the
                        ; robot continues on the path computed from the
                        ; velocities.

VelocityWt 0.1          ; range [0, 1],  Velocity weight for DWA. Velocity
                        ; refers to the linear velocity only.

NforLinVelIncrements 1  ; minumum 0,  If N is the value of this parameter, the
                        ; no of linear velocity increments of the search table
                        ; is 2*N+1 for the DWA.

NforRotVelIncrements 8  ; minumum 0,  If N is the value of this parameter, the
                        ; no of rotational velocity increments of the search
                        ; table is 2*N+1 for the DWA.

SmoothWindow 2          ; minumum 0,  Smoothing window size for DWA

ObsThreshold 0.2        ; range [0, 1],  The threshold value of the occupancy
                        ; grid to consider as occupied for path planning.

SplineDegree 3          ; minumum 1,  Degree of the B-Splines used to smooth
                        ; local path.

NumSplinePoints 5       ; minumum 1,  The number of points that will be used
                        ; to subdivide the look ahead in the local path which
                        ; will then serve as the knots for the spline to form
                        ; over.

OneWayCost 10           ; Cost of going wrong way on the one way areas. A high
                        ; value is recommended for maps with long passages
                        ; which makes loops.

CenterAwayCost 10       ; Cost of being away from the central spine of the one
                        ; way areas. A zero would cause it to not consider
                        ; centering at all.

Resistance 2            ; range [1, 32767],  Cost of traversing a cell in
                        ; restrictive areas and lines. The normal cost for
                        ; regular non-restrictive cell is 1.

StallRecoverSpeed 150   ; minimum 1,  Speed at which to back away when
                        ; stalled.

StallRecoverDuration 50 ; minumum 1,  Cycles of operation to move when
                        ; recovering from stall.

StallRecoverRotation 45 ; minimum 1,  Amount of rotation when recovering
                        ; (degrees).

DrivingTransVelMax 0    ; minimum 0,  Maximum forward translational velocity
                        ; (0 means use default)

DrivingTransNegVelMax 0 ; minimum 0,  Maximum backwards translational velocity
```

```
                         ; (0 means use default)

DrivingTransAccel 0      ; minimum 0,  Translational acceleration (0 means use
                         ; default)

DrivingTransDecel 0      ; minimum 0,  Translational deceleration (0 means use
                         ; default)

DrivingRotVelMax 0       ; minimum 0,  Maximum rotational velocity (0 means use
                         ; default)

DrivingRotAccel 0        ; minimum 0,  Rotational acceleration (0 means use
                         ; default)

DrivingRotDecel 0        ; minimum 0,  Rotational deceleration (0 means use
                         ; default)
```