FRC LabVIEW Trajectory Library – VI Implementation List

Revision 1.4 6/30/2020 – added other useful WPILIB functions

This documents which Java/C++ WPILIB routines have been duplicated in LabVIEW, and which ones are not needed (for example because all that is needed is a cluster unpack function), and what isn't done....yet...

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*
VI / CTL Totals	*380*	*368*	*132*	*352*	*180*

Doc completed Pct
96.84%
Optimization Pct
47.37%

Optimize legend: S = Subroutine, I = Inline, X = reviewed, nothing done. (In some cases, after sufficient debug and use, additional optimizations could be considered.)

'============
BASE
'============

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
LINEAR FILTER	X	X		X	SI	LinearFilter_Calculate.vi		
	X	X	X	X	X	LinearFilter_CutoffFrequency.vi		
	X	X	X	X	I	LinearFilter_Execute.vi		Labview style helper
	X	X		X	X	LinearFilter_HighPass.vi		
	X	X	X	X	X	LinearFilter_HighPassBW1.vi		
	X	X	X	X	X	LinearFilter_HighPassBW2.vi		
	X	X	X	X	X	LinearFilter_LowPassBW1.vi		
	X	X	X	X	X	LinearFilter_LowPassBW2.vi		
	X	X		X	X	LinearFilter_MovingAverage.vi		
	X	X		X	I	LinearFilter_New.vi		
	X	X		X	SI	LinearFilter_Reset.vi		
	X	X	X	X	SI	LinearFilter_ResetToValue.vi		
	X	X		X	X	LinearFilter_SinglePoleIIR.vi		
	X	X	X	X	X	LinearFilter_TimeConst.vi		

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
MEDIAN FILTER	X	X		X	X	MedianFilter_Calculate.vi		
	X	X	X	X		MedianFilter_Execute.vi		Labview style helper
	X	X		X	SI	MedianFilter_New.vi		
	X	X		X	SI	MedianFilter_Reset.vi		
	X	X	X	X	SI	MedianFilter_ResetToValue.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SLEW RATE FILTER	X	X		X		SlewRateLimiter_Calculate.vi		
	X	X	X	X		SlewRateLimiter_Close.vi		
	X	X	X	X		SlewRateLimiter_Execute.vi		Labview style helper
	X	X	X	X	SI	SlewRateLimiter_GetRate.vi		
	X	X		X		SlewRateLimiter_New.vi		
	X	X		X		SlewRateLimiter_NewInitialZero.vi		
	X	X		X		SlewRateLimiter_Reset.vi		
	X	X		X	SI	SlewRateLimiter_SetRate.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TIMER	X	X	X	X		Timer_Close.vi		releases semaphore
	X	X		X		Timer_Get.vi		
	X	X	X	X		Timer_GetAndReset.vi		
	X	X	X	No		Timer_GetInternal.vi		Internal (private) only
	X	X		X		Timer_HasPeriodPassed.vi		
	X	X	X	X		Timer_HasPeriodPassedOnce.vi		
	X	X		X		Timer_New.vi		
	X	X		X		Timer_Reset.vi		
	X	X	X	No		Timer_ResetInternal		Internal (private) only
	X	X		X		Timer_Start.vi		
	X	X		X		Timer_Stop.vi		
	X	X	X	No		Timer_StopInternal.vi		Internal (private) only

'============
CONTROLLER
'============

`

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
ARM FF	X	X		X		ArmFF_Calculate.vi		
	X	X		X		ArmFF_CalculateVelocityOnly.vi		
	X	X		X		ArmFF_MaxAchieveAccel.vi		
	X	X		X		ArmFF_MaxAchieveVelocity.vi		
	X	X		X		ArmFF_MinAchieveAccel.vi		
	X	X		X		ArmFF_MinAchieveVelocity.vi		
	X	X		X		ArmFF_New.vi		
	X	X		X		ArmFF_New_ZeroGravity.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
ELEV FF	X	X		X		ElevFF_Calculate.vi		
	X	X		X		ElevFF_CalculateVelocityOnly.vi		
	X	X		X		ElevFF_MaxAchieveAccel.vi		
	X	X		X		ElevFF_MaxAchieveVelocity.vi		
	X	X		X		ElevFF_MinAchieveAccel.vi		
	X	X		X		ElevFF_MinAchieveVelocity.vi		
	X	X		X		ElevFF_New.vi		
	X	X		X		ElevFF_New_ZeroAccel.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
PID CONTROLLER	X	X	X	X		PIDController_AdvCalculate_FF_Sp_Pv.vi		Advanced PID
	X	X	X	X		PIDController_AdvCalculate_FF_Sp_Pv_Per.vi		Advanced PID
	X	X	X	X		PIDController_AdvExecute.vi		Labview style helper. Advanced PID
	X	X		X		PIDController_AtSetpoint.vi		
	X	X		X		PIDController_Calculate_PV.vi		
	X	X		X		PIDController_Calculate_SP_PV.vi		
	X	X		X		PIDController_DisableContinousInput.vi		
	X	X		X		PIDController_EnableContinousInput.vi		
	X	X	X	X		PIDController_Execute.vi		Labview style helper
	X	X		X		PIDController_GetContinuousError.vi		
	X	X		X		PIDController_GetPeriod.vi		
	X	X		X		PIDController_GetPID.vi		
	X	X		X		PIDController_GetPositionError.vi		
	X	X		X		PIDController_GetSetpoint.vi		
	X	X		X		PIDController_GetVelocityError.vi		
	X	X		X		PIDController_New.vi		
	X	X		X		PIDController_NewPeriod.vi		
	X	X		X		PIDController_Reset.vi		
	X	X		X		PIDController_SetD.vi		
	X	X	X	X		PIDController_SetDerivativeFilter.vi		Advanced PID
	X	X	X	No		PIDController_SetFeedForward.vi		Advanced PID, Obsolete – DELETE
	X	X	X	No		PIDController_SetFFGain.vi		Advanced PID, Obsolete – DELETE
	X	X		X		PIDController_SetI.vi		
	X	X		X		PIDController_SetInputRange.vi		
	X	X		X		PIDController_SetIntegratorRange.vi		
	X	X	X	X		PIDController_SetOutputLimits.vi		Advanced PID
	X	X		X		PIDController_SetP.vi		
	X	X	X	X		PIDController_SetPeriod.vi		
	X	X		X		PIDController_SetPID.vi		
	X	X	X	X		PIDController_SetPIDF.vi		Advanced PID
	X	X		X		PIDController_SetSetpoint.vi		
	X	X		X		PIDController_SetTolerance.vi		
	X	X		X		PIDController_SetTolerancePandV.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
PROFILED PID CONTROLLER	X	X		X		ProfiledPIDController_AtGoal.vi		
	X	X		X		ProfiledPIDController_AtSetpoint.vi		
	X	X		X		ProfiledPIDController_Calculate_Meas.vi		
	X	X		X		ProfiledPIDController_Calculate_Meas_Goal.vi		
	X	X		X		ProfiledPIDController_Calculate_Meas_StateGoal.vi		
	X	X		X		ProfiledPIDController_Calculate_Meas_StateGoal_TrapCnsrt.vi		
	X	X		X		ProfiledPIDController_DisableContInput.vi		
	X	X		X		ProfiledPIDController_EnableContInput.vi		
	X	X		X		ProfiledPIDController_GetGoal.vi		
	X	X		X		ProfiledPIDController_GetPeriod.vi		
	X	X	X	X		ProfiledPIDController_GetPID.vi		WPILIB has separate getters.
	X	X		X		ProfiledPIDController_GetPositionError.vi		
	X	X		X		ProfiledPIDController_GetSetpoint.vi		
	X	X		X		ProfiledPIDController_GetVelocityError.vi		
	X	X		X		ProfiledPIDController_New.vi		
	X	X		X		ProfiledPIDController_NewPeriod.vi		
	X	X		X		ProfiledPIDController_Reset.vi		
	X	X		X		ProfiledPIDController_Reset_PosOnly.vi		
	X	X		X		ProfiledPIDController_Reset_PosVel.vi		
	X	X		X		ProfiledPIDController_SetConstraints.vi		
	X	X		X		ProfiledPIDController_SetGoal.vi		
	X	X		X		ProfiledPIDController_SetGoal_PosOnly.vi		
	X	X		X		ProfiledPIDController_SetIntegratorRange.vi		
	X	X		X		ProfiledPIDController_SetPID.vi		
	X	X		X		ProfiledPIDController_SetTolerance_PosOnly.vi		
	X	X		X		ProfiledPIDController_SetTolerance_PosVel.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
RAMSETE	X	X		X	SI	Ramsete_New.vi	new	
	X	X		X	SI	Ramsete_New_B_Z.vi	new(b, zeta)	
	X	X		X	X	Ramsete_Calculate.vi	calculate	
	X	X		X	X	Ramsete_Calculate_Trajectory.vi	calculate_trajectory	
	X	X		X	SI	Ramsete_AtReference.vi	AtReference	
	X	X		X	SI	Ramsete_SetEnabled.vi	SetEnabled	
	X	X		X	SI	Ramsete_SetTolerance.vi	SetTolerance	
	X	X		X	X	Ramsete_SINC.vi	sinc	internal
	X	X	X	X	X	Ramsete_Diff_DO_Eng.vi		
	X	X	X	X	X	Ramsete_Diff_DO_SI.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SIMPLE MOTOR FEEDFORWARD	X	X		X	SI	SimpleMotorFF_New.vi	public SimpleMotorFeedforward(double ks, double kv, double ka)	
							public SimpleMotorFeedforward(double ks, double kv)	
	X	X		X	SI	SimpleMotorFF_Calculate.vi	public double calculate(double velocity, double acceleration)	
	X	X		X	SI	SimpleMotorFF_CalculateVelocityOnly.vi	public double calculate(double velocity)	
	X	X		X	X	SimpleMotorFF_MaxAchieveVel.vi	public double maxAchievableVelocity(double maxVoltage, double acceleration)	
	X	X		X	X	SimpleMotorFF_MinAchieveVel.vi	public double minAchievableVelocity(double maxVoltage, double acceleration)	
	X	X		X	X	SimpleMotorFF_MaxAchieveAccel.vi	public double maxAchievableAcceleration(double maxVoltage, double velocity)	
	X	X		X	X	SimpleMotorFF_MinAchieveAccel.vi	public double minAchievableAcceleration(double maxVoltage, double velocity)	

'============
GEOMETRY
'============

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
POSE							pose2d new( )	can use cluster constant
	X	X		X	SI	Pose_New_TRRO.vi	pose2d new(translation2d, rotation2d)	
	X	X		X	SI	Pose_New.vi	pose2d new(double x, double y, rotation2d)	
	X	X		X	SI	Pose_Plus.vi	pose2d plus(transform2d other)	
	X	X		X	SI	Pose_Minus.vi	transform2d minus(pose2d other)	
	X	X		X	SI	Pose_getTranslation.vi	translation2d getTranslation()	can also use cluster unpack
	X	X		X	SI	Pose_getRotation.vi	rotation2d getRotation()	can also use cluster unpack
	X	X		X	SI	Pose_TransformBy.vi	pose2d transformby(transform2d other)	
	X	X		X	SI	Pose_RelativeTo.vi	pose2d relativeto(pose2d other)	
	X	X		X	X	Pose_Exp.vi	pose2d exp(twist2d twist)	
	X	X		X	X	Pose_Log.vi	twist2d log(pose2d end)	
	X	X		X	SI	Pose_Equals.VI	boolean equals(other obj)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
ROTATION							rotation2d new()	can use cluster constant
	X	X		X	SI	Rotation_CreateAngle.vi	rotation2d new(double value)	
	X	X		X	SI	Rotation_CreateXY.vi	rotation2d new(double x, double y)	
							rotation2d fromDegrees(double degrees)	convert to radians then create
	X	X		X	SI	Rotation_Plus.vi	rotation2d plus(rotation2d other)	
	X	X		X	SI	Rotation_Minus.vi	rotation2d minus(rotation2d other)	
	X	X		X	SI	Rotation_UnaryMinus.vi	rotation2d unaryminus( )	
	X	X		X	SI	Rotation_Times.vi	rotation2d times(double scalar)	
	X	X		X	SI	Rotation_RotateBy.vi	rotation2d rotateby(rotation2d other)	
	X	X		X	SI	Rotation_GetRadians.VI	double getRadians()	use cluster unpack
							double getDegrees()	use cluster unpack, then convert to degree
	X	X		X	SI	Rotation_GetCos.VI	double getCos()	use cluster unpack
	X	X		X	SI	Rotation_GetSin.VI	double getSin()	use cluster unpack
	X	X		X	SI	Rotation_GetTan.VI	double getTan()	can calculate
	X	X		X	SI	Rotation_Equals.vi	boolean equals(rotation2d other)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRANSFORM	X	X		X	SI	Transform_Create_PosePose.vi	transform2d new(pose2d, pose2d)	
	X	X		X	SI	Transform_Create_TransRot.vi	transform2d new(translation2d, rotation2d)	
							transform2d new( )	can use cluster constant
	X	X		X	SI	Transform_Times.vi	transform2d times(double scalar)	
	X	X		X	SI	Transform_GetTranslation.VI	translation2d getTranslation()	use cluster unpack
	X	X		X	SI	Transform_GetRotation.VI	rotation2d getRotation()	use cluster unpack
	X	X		X	SI	Transform_Inverse.vi	transform inverse()	new
	X	X		X	SI	Transform_Equals.VI	boolean equals(other transform2d)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRANSLATION							translation2d new()	can use cluster constant
	X	X		X	SI	Translation_Create.vi	translation2d new(double x, double y)	
	X	X		X	SI	Translation_GetDistance.vi	double getDistance(translation2d other)	
	X	X		X	SI	Translation_GetX.VI	double getX()	can use cluster unpack
	X	X		X	SI	Translation_GetY.VI	double getY()	can use cluster unpack
	X	X	X	X	SI	Translation_GetXY.VI		
	X	X		X	SI	Translation_GetNorm.VI	double getNorm()	can use cluster unpack
	X	X		X	SI	Translation_RotateBy.vi	translation2d rotateBy(rotation2d other)	
	X	X		X	SI	Translation_Plus.vi	translation2d plus(translation2d other)	
	X	X		X	SI	Translation_Minus.vi	translation2d minus(translation2d other)	
	X	X		X	SI	Translation_UnaryMinus.vi	translation2d unaryminus( )	
	X	X		X	SI	Translation_Times.vi	translation2d times(double scalar)	
							translation2d div(double scalar)	can multiply by 1/scalar
	X	X		X	SI	Translation_Equals.vi	boolean equals(translation other)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TWIST	X	X		X	SI	Twist_Create.vi	twist new(x, y, theta)	
	X	X		X	SI	Twist_Equals.VI	boolean equals(obj other)	
	X	X	X	X	SI	Twist_GetAll.VI		

'============
KINEMATICS
'============

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
CHASSIS SPEEDS							chassisspeeds new ()	can use cluster constant
	X	X		X	SI	ChassisSpeeds_New.vi	chassisspeeds new (double xvel, double yvel, double angvel)	
	X	X		X	SI	ChassisSpeeds_FromFieldRelativeSpeeds.VI	chassisspeeds fromFieldRelativeSpeeds(double x, double y, double angvel, rotation2d robotangle)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
DIFFERENTIAL DRIVE KINEMATICS	X	X		X	I	DiffKinematics_New.vi	diffDriveKine new(double trackWidth)	
	X	X		X	X	DiffKinematics_toChassisSpeed.vi	chassisSpeeds toChassisSpeeds(diffDrWheelSpeeds)	
	X	X		X	SI	DiffKinematics_toWheelSpeed.vi	diffDriveWheelSpeed toWheelSpeeds(chassisSpeeds)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
DIFFERENTIAL DRIVE ODOMETRY							diffDrOdom new(rotation gyro, pose initial)	
							diffDrOdom new(rotation gyro)	
							void resetPosition(pose2d, rotation2d)	incorporated into “update”
							pose2d getPoseMeters()	
	X	X		X	X	DiffOdometry_Update.vi	pose2d update(rotation2d gyro, double leftdist, double right dist)	Incorporates enhanced reset

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
DIFFERENTIAL DRIVE WHEEL SPEEDS							diffDrWheelSpeeds new()	
							diffDrWheelSpeeds new(double leftVel, double rightVel)	
	X	X		X	X	DiffWheel_Normalize.vi	void normalize(double maxVel)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
MECANUM DRIVE KINEMATICS	X	X		X	I	MecaKinematics_New.vi		
	X	X		X	X	MecaKinematics_SetInverseKinematics.vi		
	X	X		X	X	MecaKinematics_ToChassisSpeeds.vi		
	X	X		X	X	MecaKinematics_ToWheelSpeeds.vi		
	X	X		X	X	MecaKinematics_ToWheelSpeedsZeroCenter.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
MECANUM DRIVE MOTOR VOLTAGE								
nothing done								

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
MECANUM DRIVE ODOMETRY	X	X		X		MecaOdometry_New.vi		
	X	X		X		MecaOdometry_NewDefaultPose.vi		
	X	X		X		MecaOdometry_GetPose.vi		
	X	X		X		MecaOdometry_Reset.VI		
	X	X		X		MecaOdometry_Update.vi		
	X	X		X		MecaOdometry_UpdateWithTime.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
MECANUM DRIVE WHEEL SPEEDS	X	X		X	SI	MecaWheel_New.Vi	public MecanumDriveWheelSpeeds(double frontLeftMetersPerSecond, double frontRightMetersPerSecond, double rearLeftMetersPerSecond, double rearRightMetersPerSecond)	
	X	X		X	X	MecaWheel_Normalize.vi	public void normalize(double attainableMaxSpeedMetersPerSecond)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SWERVE DRIVE KINEMATICS							public SwerveDriveKinematics(Translation2d... wheelsMeters)	variable parameters (replace with array and “4” calls)
	X	X	X	X		SwerveKinematics_NewX.VI		uses array as input
	X	X	X	X		SwerveKinematics_New4.VI		For 4 module drives
	X	X		X		SwerveKinematics_ToSwerveModuleStates.VI	public SwerveModuleState[] toSwerveModuleStates(ChassisSpeeds chassisSpeeds, Translation2d centerOfRotationMeters)	
	X	X		X		SwerveKinematics_ToSwerveModuleStatesZeroCenter.VI	public SwerveModuleState[] toSwerveModuleStates(ChassisSpeeds chassisSpeeds)	
							public ChassisSpeeds toChassisSpeeds(SwerveModuleState... wheelStates)	variable parameters (replace with array and “4” calls)
	X	X	X	X		SwerveKinematics_ToChassisSpeedsX.VI		uses array as input
	X	X	X	X		SwerveKinematics_ToChassisSpeeds4.VI		For 4 module drives
	X	X	X	X		SwerveKinematics_NormalizeWheelSpeeds.vi	public static void normalizeWheelSpeeds(SwerveModuleState[] moduleStates, double attainableMaxSpeedMetersPerSecond)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SWERVE DRIVE ODOMETRY	X	X		X		SwerveOdometry_New.VI	public SwerveDriveOdometry(SwerveDriveKinematics kinematics, Rotation2d gyroAngle, Pose2d initialPose)	
	X	X		X		SwerveOdometry_NewZeroCenter.VI	public SwerveDriveOdometry(SwerveDriveKinematics kinematics, Rotation2d gyroAngle)	
	X	X		X		SwerveOdometry_ResetPosition.VI	public void resetPosition(Pose2d pose, Rotation2d gyroAngle)	
	X	X		X		SwerveOdometry_GetPosition.VI	public Pose2d getPoseMeters()	
							public Pose2d updateWithTime(double currentTimeSeconds, Rotation2d gyroAngle, SwerveModuleState... moduleStates)	variable parameters (replace with array and "4" calls)
	X	X	X	X		SwerveOdometry_UpdateWithTimeX.VI		uses array as input
	X	X	X	X		SwerveOdometry_UpdateWithTime4.VI		For 4 module drives
							public Pose2d update(Rotation2d gyroAngle, SwerveModuleState... moduleStates)	variable parameters (replace with array and "4" calls)
	X	X	X	X		SwerveOdometry_UpdateX.VI		uses array as input
	X	X	X	X		SwerveOdometry_Update4.VI		For 4 module drives

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SWERVE DRIVE MODULE STATE	X	X		X	SI	SwerveModuleState_New.vi	public SwerveModuleState(double speedMetersPerSecond, Rotation2d angle)	
	X	X		X	SI	SwerveModuleState_CompareTo.vi	public int compareTo(SwerveModuleState o)	

'============
SPLINE
'============

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
CUBIC HERMITE SPLINE	X	X		X		CubicHermiteSpline_New.vi	public CubicHermiteSpline(double[] xInitialControlVector, double[] xFinalControlVector, double[] yInitialControlVector, double[] yFinalControlVector)	
							protected SimpleMatrix getCoefficients()	not needed, use cluster unpack
	X	X		X		CubicHermiteSpline_makeHermiteBasis.vi	private SimpleMatrix makeHermiteBasis()	
	X	X		X		CubicHermiteSpline_getControlVectorFromArrays.vi	private SimpleMatrix getControlVectorFromArrays(double[] initialVector, double[] finalVector)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
POSE WITH CURVATURE	X	X		X	SI	PoseWithCurve_New.vi	public PoseWithCurvature(Pose2d poseMeters, double curvatureRadPerMeter)	

						public PoseWithCurvature()	can use cluster constant
						public Pose2d poseMeters	not needed, use cluster unpack
						public double curvatureRadPerMeter..	not needed, use cluster unpack

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
QUINTIC HERMITE SPLINE	X	X		X		QuinticHermiteSpline_New.vi	public QuinticHermiteSpline(double[] xInitialControlVector, double[] xFinalControlVector, double[] yInitialControlVector, double[] yFinalControlVector)	
							protected SimpleMatrix getCoefficients()	not needed, use cluster unpack
	X	X		X		QuinticHermiteSpline_makeHermiteBasis.vi	private SimpleMatrix makeHermiteBasis()	
	X	X		X		QuinticHermiteSpline_getControlVectorFromArrays.vi	private SimpleMatrix getControlVectorFromArrays(double[] initialVector, double[] finalVector)	

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
SPLINE (Abstract class)							Spline(int degree)	
	X	X		X		Spline_getPoint.vi	public PoseWithCurvature getPoint(double t)	
							public static class ControlVector	
							public ControlVector(double[] x, double[] y)	implemented as data structure

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
SPLINE HELPER	X	X		X		SplineHelp_GetCubicCtrlVectorsFromWayPts.vi	public static Spline.ControlVector[] getCubicControlVectorsFromWaypoints(Pose2d start, Translation2d[] interiorWaypoints, Pose2d end)	
	X	X		X		SplineHelp_GetQuinticCtrlVectorsFromWayPts.vi	public static List<Spline.ControlVector> getQuinticControlVectorsFromWaypoints(List<Pose2d> waypoints)	
	X	X		X		SplineHelp_getCubicSplinesFromControlVectors.vi	public static CubicHermiteSpline[] getCubicSplinesFromControlVectors(Spline.ControlVector start, Translation2d[] waypoints, Spline.ControlVector end)	
	X	X	X	No		SplineHelp_GetCubicSpline_Calc1.vi		internal
	X	X	X	No		SplineHelp_GetCubicSpline_Calc2.vi		internal
	X	X	X	No		SplineHelp_GetCubicSpline_Calc3.vi		internal
	X	X		X		SplineHelp_getQuinticSplinesFromControlVectors.vi	public static QuinticHermiteSpline[] getQuinticSplinesFromControlVectors(Spline.ControlVector[] controlVectors)	
	X	X		No		SplineHelp_ThomasAlgorithm.vi	private static void thomasAlgorithm(double[] a, double[] b, double[] c, double[] d, double[] solutionVector)	internal
	X	X		X	SI	SplineHelp_GetCubicCtrlVector.vi	private static Spline.ControlVector getCubicControlVector(double scalar, Pose2d point)	
	X	X		X	SI	SplineHelp_GetQuinticCtrlVector.vi	private static Spline.ControlVector getQuinticControlVector(double scalar, Pose2d point)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
SPLINE PARAMETERIZER	X	X		X		SplineParam_Spline.vi	public static List<PoseWithCurvature> parameterize(Spline spline)	
	X	X		X		SplineParam_Spline_T0_T1.vi	public static List<PoseWithCurvature> parameterize(Spline spline, double t0, double t1)	
	X	X	X	No		SplineParam_StackGet.vi		internal
	X	X	X	No		SplineParam_StackPop.vi		internal
	X	X	X	No		SplineParam_StackPush.vi		internal

'============
TRAJECTORY
'============

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY	X	X		X	SI	Trajectory_New.vi	public Trajectory(final List<State> states)	
							public Pose2d getInitialPose()	can use cluster unpack, array index
							public double getTotalTimeSeconds()	not needed, use unpack
							public List<State> getStates()	not needed, use unpack
	X	X		X		Trajectory_Sample.vi	public State sample(double timeSeconds)	
	X	X	X	X		Trajectory_SampleReverse.vi		Sample in reverse order. Negate sample.
	X	X		X		Trajectory_TransformBy.vi	public Trajectory transformBy(Transform2d transform)	
	X	X		X		Trajectory_RelativeTo.vi	public Trajectory relativeTo(Pose2d pose)	
	X	X		No	SI	Trajectory_lerp_double.vi	private static double lerp(double startValue, double endValue, double t)	internal
	X	X		No	SI	Trajectory_lerp_Pose.vi	private static Pose2d lerp(Pose2d startValue, Pose2d endValue, double t)	internal

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY_STATE							public State()	
	X	X		X	SI	TrajectoryState_New.vi	public State(double timeSeconds, double velocityMetersPerSecond, double accelerationMetersPerSecondSq, Pose2d poseMeters, double curvatureRadPerMeter)	
	X	X		X		TrajectoryState_Interpolate.vi	State interpolate(State endValue, double i)	
							boolean equals(other obj)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY CONFIG	X	X		X	SI	TrajectoryConfig_Create.vi	public TrajectoryConfig(double maxVelocityMetersPerSecond, double maxAccelerationMetersPerSecondSq)	
							public TrajectoryConfig addConstraint(TrajectoryConstraint constraint)	Implemented differently, can't duplicate.
							public TrajectoryConfig addConstraints(List<? extends TrajectoryConstraint> constraints)	Implemented differently, can't duplicate.
	X	X		X	SI	TrajectoryConfig_setKinematicsDiffDrive.vi	public TrajectoryConfig setKinematics(DifferentialDriveKinematics kinematics)	
	X	X		X	SI	TrajectoryConfig_setKinematicsMecanumfDrive.vi	public TrajectoryConfig setKinematics(MecanumDriveKinematics kinematics)	
	X	X		X	SI	TrajectoryConfig_setKinematicsSwerveDrive.vi	public TrajectoryConfig setKinematics(SwerveDriveKinematics kinematics)	
							public double getStartVelocity()	can use cluster unpack
							public TrajectoryConfig setStartVelocity(double startVelocityMetersPerSecond)	
							public double getEndVelocity()	can use cluster unpack
							public TrajectoryConfig setEndVelocity(double endVelocityMetersPerSecond)	
							public double getMaxVelocity()	can use cluster unpack
							public double getMaxAcceleration()	can use cluster unpack
							public List<TrajectoryConstraint> getConstraints()	Implemented differently, can't duplicate.
							public boolean isReversed()	can use cluster unpack
	X	X		X	SI	TrajectoryConfig_setReversed.vi	public TrajectoryConfig setReversed(boolean reversed)	
	X	X	X	X	SI	TrajectoryConfig_setCentripetalAccel.vi		
	X	X	X	X	SI	TrajectoryConfig_setVoltageDiffDrive.vi		
							NOTE ADD OTHER "SET" ROUTINES FOR OTHER CONTRAINTS HERE, SINCE NEW CONTRAINTS ARE SPECIFIC AND NOT GENERIC.	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY GENERATE	X	X		X		TrajectoryGenerate_Make_Cubic_CtrlVect.vi	public static Trajectory generateTrajectory(Spline.ControlVector initial, List<Translation2d> interiorWaypoints, Spline.ControlVector end, TrajectoryConfig config)	uses cubic splines
	X	X		X		TrajectoryGenerate_Make_Cubic.vi	public static Trajectory generateTrajectory(Pose2d start, List<Translation2d> interiorWaypoints, Pose2d end, TrajectoryConfig config)	uses cubic splines
	X	X		X		TrajectoryGenerate_Make_Quintic_CtrlVect.vi	public static Trajectory generateTrajectory(ControlVectorList controlVectors, TrajectoryConfig config)	uses quintic splines
	X	X		X		TrajectoryGenerate_Make_Quintic.vi	public static Trajectory generateTrajectory(List<Pose2d> waypoints, TrajectoryConfig config)	uses quintic splines
	X	X		X		TrajectoryGenerate_splinePointsFromSplines.vi	public static List<PoseWithCurvature> splinePointsFromSplines(Spline[] splines)	

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY GENERATE (Control Vector)							public ControlVectorList(int initialCapacity)	may not need, just data
							public ControlVectorList()	may not need, just data
							public ControlVectorList(Collection<? extends Spline.ControlVector> collection)	may not need, just data

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY PARAMETERIZE	X	X		X		TrajectoryParam_timeParam.vi	public static Trajectory timeParameterizeTrajectory(List<PoseWithCurvature> points. List<TrajectoryConstraint> constraints, double startVelocityMetersPerSecond, double endVelocityMetersPerSecond, double maxVelocityMetersPerSecond, double maxAccelerationMetersPerSecondSq, boolean reversed)	
	X	X		No		TrajectoryParam_enforceAccel.vi	private static void enforceAccelerationLimits(boolean reverse, List<TrajectoryConstraint> constraints, ConstrainedState state)	This routines needs to be changed when new constraints are added.
	X	X	X	No		TrajectoryParam_calcStuffFwd.vi		
	X	X	X	No		TrajectoryParam_calcStuffRev.vi		
	X	X	X	No		TrajectoryParam_enforceVelocity.vi		This routines needs to be changed when new constraints are added.

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY PARAMETERIZE CONSTRAINED STATE	X	X		X		ConstrainedState_New.vi	ConstrainedState(PoseWithCurvature pose, double distanceMeters, double maxVelocityMetersPerSecond, double minAccelerationMetersPerSecondSq, double maxAccelerationMetersPerSecondSq)	
							ConstrainedState()	
	X	X	X	X		ConstrainedState_SetMaxAccel.vi		
	X	X	X	X		ConstrainedState_SetMinAccel.vi		
	X	X	X	X		ConstrainedState_SetVelAccel.vi		
	X	X	X	X		ConstrainedState_SetVelocity.vi		

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY UTIL	X	X		X		TrajectoryUtil_fromPathWeaverJSON.vi	public static Trajectory fromPathweaverJson(Path path)	

						public static void toPathweaverJson(Trajectory trajectory, Path path)	
						public static Trajectory deserializeTrajectory(String json)	
						public static String serializeTrajectory(Trajectory trajectory)	

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
TRAPEZOID PROFILE	X	X		X		TrapProfConstraint_New.vi		
	X	X		X		TrapProfile_Calculate.vi		
	X	X		No		TrapProfile_Direct.vi		Private, remove from menu
	X	X	X	X		TrapProfile_Execute.vi		
	X	X		X		TrapProfile_IsFinished.vi		
	X	X		X		TrapProfile_New.vi		
	X	X		X		TrapProfile_New_DefInitial.vi		
	X	X		No		TrapProfile_ShouldFlipAcceleration.vi		Private, remove from menu
	X	X		X		TrapProfile_TimeLeftUntil.vi		
	X	X		X		TrapProfile_TotalTime.vi		
	X	X		X		TrapProfState_Equals.vi		
	X	X		X		TrapProfState_New.vi		

'============
TRAJECTORY CONSTRAINT
'============

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
CENTRIPETAL ACCELERATION CONSTRAINT	X	X		X		CentripetalAccelConstraint_getMaxVelocity.vi	public double getMaxVelocityMetersPerSecond(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X		CentripetalAccelConstraint_getMinMaxAccel.vi	public MinMax getMinMaxAccelerationMetersPerSecondSq(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X	SI	CentripetalAccelConstraint_New.vi	public CentripetalAccelerationConstraint(double maxCentripetalAccelerationMetersPerSecondSq)	Can use cluster pack for now

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
DIFF DRIVE KINEMATIC CONSTRAINT	X	X		X		DiffDriveKinematicsConstraint_getMaxVelocity.vi	public double getMaxVelocityMetersPerSecond(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X		DiffDriveKinematicsConstraint_getMinMaxAccel.vi	public MinMax getMinMaxAccelerationMetersPerSecondSq(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	

	X	X		X	SI	DiffDriveKinematicsConstraint_New.vi	public DifferentialDriveKinematicsConstraint(final DifferentialDriveKinematics kinematics, double maxSpeedMetersPerSecond)	

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
DIFF DRIVE VOLTAGE CONSTRAINT	X	X		X		DiffDriveVoltageConstraint_getMaxVelocity.vi	public double getMaxVelocityMetersPerSecond(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X		DiffDriveVoltageConstraint_getMinMaxAccel.vi	public MinMax getMinMaxAccelerationMetersPerSecondSq(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	Code updated to match 2/2020 library update.
	X	X		X	SI	DiffDriveVoltageConstraint_New.vi	public DifferentialDriveVoltageConstraint(SimpleMotorFeedforward feedforward, DifferentialDriveKinematics kinematics, double maxVoltage)	Can use cluster pack for now

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
MECANUM DRIVE KINEMATICS CONSTRAINT	X	X		X	SI	MecaDriveKinematicsConstraint_New.vi		
	X	X		X		MecaDriveKinematicsConstraint_getMaxVelocity.vi		
	X	X		X		MecaDriveKinematicsConstraint_getMinMaxAccel.vi		

	Implemented	*Documented*	*Not WPILIB*	*Menu Item*	*Execution Optimized*	VI Name	Function Prototype	Notes
SWERVE DRIVE KINEMATICS CONSTRAINT	X	X		X		SwerveDriveKinematicsConstraint_getMaxVelocity.vi	public double getMaxVelocityMetersPerSecond(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X		SwerveDriveKinematicsConstraint_getMinMaxAccel.vi	public MinMax getMinMaxAccelerationMetersPerSecondSq(Pose2d poseMeters, double curvatureRadPerMeter, double velocityMetersPerSecond)	
	X	X		X	SI	SwerveDriveKinematicsConstraint_New.vi	Newpublic SwerveDriveKinematicsConstraint(final SwerveDriveKinematics kinematics, double maxSpeedMetersPerSecond)	Can use cluster pack for now

TRAJECTORY CONSTRAINT

Interface class - nothing done (not needed)

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TRAJECTORY CONSTRAINT (Min Max)	X	X		X	SI	Constraint_MinMax_New.vi	Constraint_MinMax_New	
	X	X		X	SI	Constraint_MinMax_NewMinMax.VI	Constraint_MinMax_New	

'============
UTILITY
'============
THESE ROUTINES ARE SPECIFIC TO LABVIEW. THEY DO NOT HAVE A
JAVA / C++ WPILIB EQUIVALENT

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
UTIL	X	X	X	X		Util_Array_PoseWCurv_to_XY.vi		
	X	X	X	X	SI	Util_CalcDist.vi		
	X	X	X	X	SI	Util_GetLibraryVersion.vi		
	X	X	X	X	SI	Util_GetLibraryUsage.vi		
	X	X	X	X		Util_GetTime.vi		Once tested completely, this should be optimized!
	X	X	X	No	N/A	Util_LibraryGlobals.vi		Global Variables – no block diag.
	X	X	X	X		Util_Trajectory_Absolute_To_Relative.vi		
	X	X	X	X		Util_Trajectory_ReadFile.vi		
	X	X	X	X		Util_Trajectory_WriteFile.vi		
	X	X	X	No		Util_Trajectory_WriteFile_Config.vi		internal
	X	X	X	No		Util_Trajectory_WriteFile_OneState.vi		internal
	X	X	X	X		Util_Trajectory_WriteFile_PathFinder.vi		
	X	X	X	No		Util_Trajectory_WriteFile_PathFinderConfig.vi		internal
	X	X	X	X		Util_Trajectory_WriteFile_Pathweaver.vi		
	X	X	X	No		Util_Trajectory_WriteFile_States.vi		internal
	X	X	X	No		Util_Trajectory_WriteFile_WayPoints.vi		internal
	X	X	X	X		Util_TrajectoryState_Meters_To_Inches.vi		
	X	X	X	X		Util_TrajState_to_DiffDrive_WheelPos.vi		
	X	X	X	X		Util_Waypoint_Eng_To_SI.vi		
	X	X	X	X		Util_Waypoint_To_CubicInput.vi		
	X	X	X	X		Util_Waypoint_To_QuinticInput.vi		

'============
CONVERSIONS
'============
THESE ROUTINES ARE SPECIFIC TO LABVIEW. THEY DO NOT HAVE A
JAVA / C++ WPILIB EQUIVALENT

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
CONV	X	X	X	X	SI	Conv_AngleDegrees_Heading.vi		
	X	X	X	X	SI	Conv_AngleRadians_Heading.vi		

X	X	X	X	SI	Conv_Centimeters_Meters.vi		
X	X	X	X	SI	Conv_Deg_Radians.vi		
X	X	X	X	SI	Conv_Feet_Meters.vi		
X	X	X	X	SI	Conv_GyroDegrees_Heading.vi		
X	X	X	X	SI	Conv_Heading_AngleRadians.vi		
X	X	X	X	SI	Conv_Inches_Meters.vi		
X	X	X	X	SI	Conv_Meters_Feet.vi		
X	X	X	X	SI	Conv_Meters_Inches.vi		
X	X	X	X	SI	Conv_POSE_SI_Eng.vi		
X	X	X	X	SI	Conv_Radians_Deg.vi		
X	X	X	X	SI	Conv_Yards_Meters.vi		

'============
PATHFINDER UTIL
'============

THESE ROUTINES ARE SPECIFIC TO LABVIEW. THEY DO NOT HAVE A
JAVA / C++ WPILIB EQUIVALENT

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
PATHFINDERUTIL	X	X	X	X		PathfinderUtil_Continuous_Heading_Difference.vi		
	X	X	X	X		PathfinderUtil_OptimizeTrajectoryStates.vi		
	X	X	X	X		PathfinderUtil_ToTrajectory.vi		
	X	X	X	X		PathfinderUtil_ToTrajectoryStates.vi		

'============
TYPE DEFINITIONS
'============

	Implemented	Documented	Not WPILIB	Menu Item	Execution Optimized	VI Name	Function Prototype	Notes
TypeDef	X			X	N/A	ARM_FF.CTL		
	X	X	X	X	N/A	CHASSIS_SPEEDS.CTL		
	X	X	X	X	N/A	CONTRAINED_STATE.CTL		
	X	X	X	X	N/A	DIFF_DRIVE_KINEMATICS.CTL		
	X			X	N/A	ELEV_FF.CTL		
	X	X		X	N/A	LINEAR_FILTER.CTL		
	X	X	X	X	N/A	MECA_DRIVE_KINEMATICS.CTL		
	X	X	X	X	N/A	MECA_DRIVE_ODOMETRY.CTL		
	X	X	X	X	N/A	MECA_WHEEL_SPEEDS.CTL		
	X			X	N/A	MEDIAN_FILTER.CTL		
	X	X	X	X	N/A	PARAM_STACK.CTL		
	X	X	X	X	N/A	PARAM_STACK_ITEM.CTL		
	X			X	N/A	PID_CONTROLLER		
	X	X	X	X	N/A	POSE2D.CTL		
	X	X	X	X	N/A	POSEwCURVATURE.CTL		
	X			X	N/A	PROFILED_PID_CONTROLLER.CTL		
	X	X	X	X	N/A	RAMSETE.CTL		
	X	X	X	X	N/A	ROTATION2D.CTL		
	X	X	X	X	N/A	SIMPLE_MOTOR_FF.CTL		
	X			X	N/A	SLEW_RATE_LIMITER.CTL		

X	X	X	X	N/A	SPLINE.CTL		
X	X	X	X	N/A	SPLINE_CTRL_VECTOR.CTL		
X	X	X	X	N/A	SWERVE_DRIVE_KINEMATICS.CTL		
X	X	X	X	N/A	SWERVE_DRIVE_MODULE_STATE.CTL		
X	X	X	X	N/A	SWERVE_DRIVE_ODOMETRY.CTL		
X			X	N/A	TIMER.CTL		
X	X	X	X	N/A	TRAJ_CONFIG.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_CENTRIPETAL_ACCEL.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_DIFF_DRIVE_KINEMATICS.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_DIFF_DRIVE_VOLTAGE.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_MECA_DRIVE_KINEMATICS.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_MINMAX.CTL		
X	X	X	X	N/A	TRAJ_CONSTRAINT_SWERVE_DRIVE_KINEMATICS.CTL		
X	X	X	X	N/A	TRAJ_STATE.CTL		
X	X	X	X	N/A	TRAJECTORY.CTL		
X	X	X	X	N/A	TRANSFORM2D.CTL		
X	X	X	X	N/A	TRANSLATION2D.CTL		
X			X	N/A	TRAPEZOID_PROFILE.CTL		
X			X	N/A	TRAPEZOID_PROFILE_CONSTRAINT.CTL		
X			X	N/A	TRAPEZOID_PROFILE_STATE.CTL		
X	X	X	X	N/A	TWIST2D.CTL		
X	X	X	X	N/A	UTIL_PATHFINDER_CONFIG.CTL		
X	X	X	X	NA	UTIL_WAYPOINT.ctl		
X		X		N/A	WAYPOINTS.CTL		Delete – obsolete
X		X		N/A	X_Y_HEADINGS.CTL		Delete – obsolete