

PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES SOLDADORES BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD

Proyecto de investigación que para obtener el grado de Magister en
Automatización y Control Industrial presenta:

Juan Camilo Ramírez Henao

Director

M. eng. Leonardo Duque Muñoz



Institución Universitaria

INSTITUTO TECNOLÓGICO METROPOLITANO

FACULTAD DE INGENIERÍAS

MEDELLÍN

2013

NOTA DE ACEPTACIÓN:

Presidente del Jurado

Jurado

Jurado

TABLA DE CONTENIDO

1	MARCO TEÓRICO	22
1.1	GENERALIDADES DE LOS MANIPULADORES SERIALES	22
1.1.1	CLASIFICACIÓN CINEMÁTICA DE LOS MANIPULADORES SERIALES.....	22
1.1.2	MUÑECA Y EFECTOR FINAL	26
1.2	CINEMÁTICA MANIPULADORES SERIALES	27
1.2.1	CINEMÁTICA DIRECTA MANIPULADORES SERIALES.....	27
1.2.2	CINEMÁTICA INVERSA MANIPULADORES SERIALES	31
1.3	CINEMÁTICA DIFERENCIAL MANIPULADORES SERIALES	32
1.3.1	CALCULO JACOBIANA GEOMÉTRICA	33
1.3.2	CINEMÁTICA INVERSA DIFERENCIAL.....	34
1.3.3	MANIPULABILIDAD TRANSLACIONAL Y ROTACIONAL MANIPULADORES SERIALES .	35
1.4	DINÁMICA MANIPULADORES SERIALES.....	37
1.4.1	DINÁMICA INVERSA SEGÚN FORMULACIÓN NEWTON EULER PARA MANIPULADOR SERIAL	37
1.4.2	CONSUMO DE ENERGÍA MANIPULADORES SERIALES.....	42
1.4.3	PARÁMETROS DINÁMICOS.....	43
1.5	MÉTODO DE OPTIMIZACIÓN MEDIANTE ALGORITMO HEURÍSTICO DE KALMAN	45
1.5.1	PRINCIPIO DEL ALGORITMO HKA.....	47
1.5.2	DESARROLLO DEL ALGORITMO HKA	50
1.6	PRODUCTIVIDAD.....	51
2	MODELADO MATEMÁTICO DE ECUACIONES PARA PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA	53
2.1	CINEMÁTICA DIRECTA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA.....	53
2.2	CINEMÁTICA INVERSA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA.....	57
2.3	CINEMÁTICA DIFERENCIAL MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA.....	61
2.3.1	CÁLCULO VELOCIDADES ARTICULARES	61

2.3.2	CÁLCULO ACELERACIONES ARTICULARES	65
2.3.3	MANIPULABILIDAD TRANSLACIONAL Y ROTACIONAL MANIPULADORES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA	66
2.4	DINÁMICA MANIPULADORES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA.....	67
2.4.1	DINÁMICA INVERSA MANIPULADORES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA	67
2.4.2	CONSUMO DE ENERGÍA MANIPULADORES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA	68
2.5	ESPACIO DE TRABAJO SIMPLIFICADO EN MANIPULADORES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA.....	69
2.6	MÉTODO OPTIMIZACIÓN HKA APLICADO A LOS CRITERIOS DE CONSUMO ELÉCTRICO Y MANIPULABILIDAD DEL MANIPULADOR ANROPOMÓRFICO DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA.....	73
2.7	ALGORITMO DE PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD	78
2.7.1	VARIABLES DE ENTRADA	78
2.7.2	VARIABLES SALIDA	80
2.7.3	DIAGRAMA FLUJO SIMPLIFICADO ALGORITMO PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD	84
2.7.4	DIAGRAMA DE FLUJO SIMPLIFICADO FUNCIÓN VALIDACIÓN TRAYECTORIA	90
2.7.5	PROGRAMA DE PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD	92
3	RESULTADOS	96
3.1	MANIPULADORES USADOS CON ALGORITMO DE PLANEACIÓN DE TRAYECTORIAS	97
3.1.1	Manipulador UNIMATE PUMA 560.....	97
3.1.2	Manipulador KUKA KR5 HW ARC.....	102
3.2	TRAYECTORIA 1: ROBOTWORKS.....	105
3.3	TRAYECTORIA 2: TARRO INCLINADO	120
3.4	TRAYECTORIA 3: FIGURA LÍNEAS RECTAS	131
3.5	TRAYECTORIA 4: HÉLICE UN PASO.....	140
3.6	TRAYECTORIA 5: GABINETE	149

LISTADO DE FIGURAS

Figura 1 Mapa Conceptual	19
Figura 2 Manipulador Antropomórfico	23
Figura 3 Manipulador Esférico.....	24
Figura 4 Manipulador SCARA.....	24
Figura 5 Manipulador Cilíndrico	25
Figura 6 Manipulador Cartesiano	25
Figura 7 Muñeca esférica	26
Figura 8 Modelo cinemático general eslabones manipulador según convención DH	29
Figura 9 Manipulador de seis grados de libertad con muñeca esférica.....	31
Figura 10 Vectores necesarios para calcular la contribución de la velocidad de una articulación en la velocidad del efector final	33
Figura 11 Caracterización del Eslabón para la formulación Newton Euler	38
Figura 12 Esquema modelo eléctrico del motor (Corke P. , The Unimation Puma servo system, 1994)	44
Figura 13 Principio del algoritmo HKA	48
Figura 14 Esquema convención DH robot antropomórfico de seis grados de libertad con muñeca esférica	54
Figura 15 Esquema para cinemática inversa de robot antropomórfico de seis grados de libertad con muñeca esférica	58
Figura 16 Esquema tres primeros eslabones robot antropomórfico de seis grados de libertad con muñeca esférica.....	59
Figura 17 Sistema coordenadas cinemática diferencial y manipulabilidad	62
Figura 18 Espacio de trabajo simplificado manipulador antropomórfico seis grados de libertad y muñeca esférica.....	70
Figura 19 Esquema para cálculo de radio mayor y menor de espacio de trabajo manipulador antropomórfico seis grados de libertad y muñeca esférica	70
Figura 20 Esquema coordenadas cartesianas a partir de variables de espacio de trabajo simplificado	72
Figura 21 Diagrama de flujo funciones principales algoritmo de planeación de trayectorias	86

Figura 22 Diagrama de flujo función Opt_HKA (parte 1)	87
Figura 23 Diagrama de flujo función Opt_HKA (parte 2)	88
Figura 24 Diagrama de flujo función Opt_HKA (parte 3)	89
Figura 25 Diagrama de flujo función para validación trayectoria (parte 1)	90
Figura 26 Diagrama de flujo función para validación trayectoria (parte 2)	91
Figura 27 Página 1 programa de planeación de trayectorias.....	93
Figura 28 Página 2 programa de planeación de trayectorias.....	93
Figura 29 Página 3 programa de planeación de trayectorias.....	94
Figura 30 Página 4 programa de planeación de trayectorias.....	95
Figura 31 Página 4 programa de planeación de trayectorias.....	95
Figura 32 Esquema convención DH PUMA 560	98
Figura 33 Pistola soldadura de arco TOUGH GUN G2	101
Figura 34 Esquema convención DH KUKA KR5 ARC HW	103
Figura 35 Pistola soldadura de arco ABIROB A	104
Figura 36 Trayectoria Robotworks.....	105
Figura 37 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 con parámetros dinámicos	106
Figura 38 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 solo con parámetros cinemáticos	106
Figura 39 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot KUKA KR5 HW ARC solo con parámetros cinemáticos.....	107
Figura 40 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 con parámetros dinámicos	109
Figura 41 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 solo con parámetros cinemáticos	110
Figura 42 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot KUKA KR5 HW ARC solo con parámetros cinemáticos.....	110
Figura 43 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot PUMA 560 con parámetros dinámicos	112
Figura 44 Trayectoria Robotworks en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos.....	112
Figura 45 Trayectoria Robotworks Optimizada en Robot PUMA 560 con parámetros dinámicos.	113

Figura 46 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original Vs optimizada en Robot PUMA 560 con parámetros dinámicos.....	113
Figura 47 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot PUMA 560 solo parámetros cinemáticos.....	115
Figura 48 Trayectoria Robotworks Optimizada en Robot PUMA 560 solo parámetros cinemáticos	115
Figura 49 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos	116
Figura 50 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	118
Figura 51 Trayectoria Robotworks en Robot KUKA KR5 HW.....	118
Figura 52 Trayectoria Robotworks Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	119
Figura 53 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos..	119
Figura 54 Trayectoria Tarro Inclinado	120
Figura 55 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 con parámetros dinámicos.....	123
Figura 56 Trayectoria Tarro Inclinado en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos.....	124
Figura 57 Trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 con parámetros dinámicos	124
Figura 58 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot PUMA 560 con parámetros dinámicos.....	125
Figura 59 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 solo parámetros cinemáticos	126
Figura 60 Trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 solo parámetros cinemáticos	127
Figura 61 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos.....	127
Figura 62 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	129

Figura 63 Trayectoria Tarro Inclinado sin optimizar en Robot KUKA KR5 HW	129
Figura 64 Trayectoria Tarro Inclinado Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	130
Figura 65 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	130
Figura 66 Trayectoria Figura Líneas Rectas	132
Figura 67 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 con parámetros dinámicos	132
Figura 68 Trayectoria Figura Líneas Rectas en Robot PUMA 560	133
Figura 69 Trayectoria Figura Líneas Rectas en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos	133
Figura 70 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot PUMA 560 con parámetros dinámicos	134
Figura 71 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 solo parámetros cinemáticos.....	135
Figura 72 Trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 solo parámetros cinemáticos	135
Figura 73 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos	136
Figura 74 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos.....	138
Figura 75 Trayectoria Figura Líneas Rectas en Robot KUKA KR5 HW	138
Figura 76 Trayectoria Figura Líneas Rectas Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos.....	139
Figura 77 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	139
Figura 78 Trayectoria Hélice un Paso.....	140
Figura 79 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot PUMA 560 con parámetros dinámicos.....	141
Figura 80 Trayectoria Hélice un Paso en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos.....	141

Figura 81 Trayectoria Hélice un Paso Optimizada en Robot PUMA 560 con parámetros dinámicos	142
Figura 82 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot PUMA 560 con parámetros dinámicos.....	142
Figura 83 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot PUMA 560 solo parámetros cinemáticos	144
Figura 84 Trayectoria Hélice un Paso Optimizada en Robot PUMA 560 solo parámetros cinemáticos	144
Figura 85 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos.....	145
Figura 86 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	147
Figura 87 Trayectoria Hélice un Paso sin optimizar en Robot KUKA KR5 HW	147
Figura 88 Trayectoria Hélice un Paso Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	148
Figura 89 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	148
Figura 90 Trayectoria Gabinete	149
Figura 91 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot PUMA 560 con parámetros dinámicos.....	150
Figura 92 Trayectoria Gabinete en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos.....	150
Figura 93 Trayectoria Gabinete Optimizada en Robot PUMA 560 con parámetros dinámicos	151
Figura 94 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete original vs optimizada en Robot PUMA 560 con parámetros dinámicos.....	151
Figura 95 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos	153
Figura 96 Trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos..	153
Figura 97 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos	154
Figura 98 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	156

Figura 99 Trayectoria Gabinete sin optimizar en Robot KUKA KR5 HW	156
Figura 100 Trayectoria Gabinete Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	157
Figura 101 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos	157

LISTADO DE TABLAS

Tabla 1 Ajuste de los parámetros estándar de HKA	50
Tabla 2 Modelo cinemático basado en la convención DH para manipulador antropomórfico de seis grados de libertad con muñeca esférica	54
Tabla 3 Variables de decisión algoritmo HKA.....	74
Tabla 4 Variables entrada parámetros cinemáticos manipuladores antropomórficos de seis grados de libertad y muñeca esférica	78
Tabla 5 Variables entrada parámetros diferenciales manipuladores antropomórficos de seis grados de libertad y muñeca esférica.	79
Tabla 6 Variables entrada parámetros dinámicos manipuladores antropomórficos de seis grados de libertad y muñeca esférica.	80
Tabla 7 Variables entrada trayectoria manipuladores antropomórficos de seis grados de libertad y muñeca esférica.....	80
Tabla 8 Variables salida parámetros cinemáticos manipuladores antropomórficos de seis grados de libertad y muñeca esférica	82
Tabla 9 Variables salida parámetros diferenciales manipuladores antropomórficos de seis grados de libertad y muñeca esférica	82
Tabla 10 Variables salida parámetros dinámicos manipuladores antropomórficos de seis grados de libertad y muñeca esférica	83
Tabla 11 Variables salida optimizadas manipuladores antropomórficos de seis grados de libertad y muñeca esférica.....	84
Tabla 12 Salidas matriz que refleja comportamiento del algoritmo de optimización en k iteraciones para manipuladores antropomórficos de seis grados de libertad y muñeca esférica.....	84
Tabla 13 Modelo cinemático basado en la convención DH para manipulador PUMA 560	98
Tabla 14 Límite articulaciones PUMA 560.....	99
Tabla 15 Velocidad lineal máxima efector final PUMA 560	99
Tabla 16 Límite velocidades articulares PUMA 560.....	99
Tabla 17 Centro de masa eslabones PUMA 560.....	99
Tabla 18 Peso eslabones PUMA 560.....	100
Tabla 19 Momentos de inercia eslabones PUMA 560	100

Tabla 20 Coeficientes Viscosidad Motores Robot PUMA 560	100
Tabla 21 Constante Torque Motores Robot PUMA 560	101
Tabla 22 Constante De Fuerza Contra Electromotriz Motores Robot PUMA 560	101
Tabla 23 Resistencia Armaduras Motores Robot PUMA 560	101
Tabla 24 Relación De Transmisión Motores Robot PUMA 560	101
Tabla 25 Potencia nominal PUMA 560.....	101
Tabla 26 Datos generales pistola TOUGH GUN G2	102
Tabla 27 Centro de masa pistola TOUGH GUN G2.....	102
Tabla 28 Matriz momento de inercia pistola TOUGH GUN G2.....	102
Tabla 29 Modelo cinemático basado en la convención DH para manipulador KUKA KR5 ARC HW103	
Tabla 30 Límite articulaciones KUKA KR5 ARC HW	104
Tabla 31 Velocidad lineal máxima efector final KUKA KR5 ARC HW.....	104
Tabla 32 Límite velocidades articulares KUKA KR5 ARC HW	104
Tabla 33 Potencia nominal KUKA KR5 ARC HW	104
Tabla 34 Datos generales pistola ABIROB A	105
Tabla 35 Comparación resultados diferentes pesos en criterios de optimización para trayectoria Robotworks.	108
Tabla 36 Comparación resultados de coeficientes de ralentización en trayectoria Robotworks. .	111
Tabla 37 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot PUMA 560 con parámetros dinámicos.....	114
Tabla 38 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos	116
Tabla 39 Comparación de parámetros optimizados en trayectoria Robotworks original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos	117
Tabla 40 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos	120
Tabla 41 Comparación resultados diferentes pesos en criterios de optimización para trayectoria Tarro Inclinado.....	121
Tabla 42 Comparación resultados de coeficientes de ralentización en trayectoria Tarro Inclinado.	122
Tabla 43 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot PUMA 560 con parámetros dinámicos.....	125

Tabla 44 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos	128
Tabla 45 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos	128
Tabla 46 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos	131
Tabla 47 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot PUMA 560 con parámetros dinámicos	134
Tabla 48 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos	136
Tabla 49 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos	137
Tabla 50 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos	140
Tabla 51 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot PUMA 560 con parámetros dinámicos	143
Tabla 52 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos	145
Tabla 53 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos	146
Tabla 54 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos	149
Tabla 55 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot PUMA 560 con parámetros dinámicos	152
Tabla 56 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos	154
Tabla 57 Comparación de parámetros optimizados en trayectoria Gabinete original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos	155
Tabla 58 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos	158

RESUMEN

En el presente trabajo se desarrolla una metodología de planeación de trayectorias para manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica enfocada en la minimización del consumo eléctrico y maximización de la manipulabilidad. El trasfondo (undercurrent) de esta metodología, es garantizar un aumento en la productividad maximizando el rendimiento del manipulador.

Para lograr tal fin se propone un espacio de trabajo esférico simplificado, se plantean los modelos de cálculo de la cinemática directa, inversa y diferencial, al igual que la dinámica inversa de los manipuladores, y se integran en un algoritmo de optimización el cual tiene como base el algoritmo heurístico de Kalman.

El algoritmo de optimización se evalúa mediante el análisis del comportamiento de cinco trayectorias realizadas en los manipuladores PUMA 560 y KUKA KR5 HW ARC, simuladas en un programa desarrollado en la aplicación de Visual Basic perteneciente al software Autodesk Inventor.

El enfoque de planeación de trayectoria propuesto permite ser usado en manipuladores antropomórficos de seis grados de libertad y muñeca esférica, de los cuales se conozca sus parámetros cinemáticos y dinámicos, para generar trayectorias optimizadas desde los criterios de manipulabilidad y energía eléctrica, conservando la orientación del efector final.

Una ventaja importante que posee el método es que a partir de una trayectoria planteada en forma de coordenadas cartesianas XYZ y orientaciones del efector en ángulos tipo Euler (precesión, nutación y rotación propia), buscará una solución óptima dentro del espacio de trabajo simplificado del manipulador respetando sus límites articulares y de velocidad, permitiendo con esto tener trayectorias de entrada en las cuales solo importe la orientación de la herramienta a lo largo del recorrido, situación indispensable para el proceso de soldadura.

ABSTRACT

In the present work a path planning methodology for serial anthropomorphic manipulator with six degrees of freedom spherical wrist are developed, focused on minimizing power consumption and maximizing the manipulability. The background of this methodology is to ensure an increase of the productivity by maximizing the performance of the manipulator.

To achieve this goal a simplified spherical workspace are proposed, computational models of direct, inverse and differential kinematics, and inverse dynamics for serial anthropomorphic manipulators are developed, and integrated into an optimization algorithm based on a heuristic Kalman algorithm.

The optimization algorithm is evaluated by analyzing the behavior of five trajectories made in the PUMA 560 and KUKA KR5 HW 560 ARC manipulators, simulated in a program developed in an application of Visual Basic belongs to Autodesk Inventor software.

The proposed path planning approach allows to be used in anthropomorphic manipulators with six degrees of freedom spherical wrist, which its kinematic and dynamic parameters are known to generate optimized trajectories from manipulability criteria and electricity, preserving the orientation of the end effector.

An important advantage of this the method, is that from a trajectory presented as Cartesian coordinates type XYZ, and orientations of the end effector in Euler type angles (precession, nutation and proper rotation), it can seek a optimal solution within the simplified workspace of the manipulator, respecting their joint and speed limits, allowing it to have input trajectories in which only care the tool orientation along the path.

ANTECEDENTES

INTRODUCCIÓN

La soldadura por arco bajo gas protector, es uno de los procesos de unión de metales más comunes usados para el ensamble de estructuras metálicas en la industria automotriz, aeronáutica y naval, debido a la resistencia de su junta, la confiabilidad y bajo costo, en comparación con otros procesos de unión (Son et al., 2007). La estabilidad de este proceso de soldadura es muy sensible a los parámetros principales de soldadura, especialmente la corriente, el voltaje, velocidad de soldadura, longitud del alambre que sale de la pistola, gas de protección y longitud del arco. Un pequeño cambio en la distancia entre la pistola de soldadura y la pieza a soldar puede producir una variación considerable en la corriente y el voltaje. La corriente, el voltaje y el gas de protección influyen en el modo de transferencia del metal fundente base al componente que se está soldando, afectando la calidad de la soldadura. Si el arco eléctrico es inestable, defectos como mala penetración, cizallados y exceso de salpicaduras pueden presentarse (Pires, Loureiro, & Bölmjö, 2006).

La soldadura robótica es una de las aplicaciones más exitosas de los robots manipuladores industriales. Esto se debe principalmente a la abolición de deficiencias del proceso generadas por el factor humano al introducir robots para desempeñar tales tareas. Esto conlleva a productos más baratos debido a que la productividad y calidad aumentan, y los costos de producción y mano de obra disminuyen (Son et al., 2007). El proceso de soldadura robótica presenta muchas ventajas comparado con la soldadura manual, debido a que los problemas técnicos presentes en la soldadura manual, como la variación de la distancia del material de aporte a la pieza, y desplazamientos a velocidad constante de la pistola de soldadura pueden ser resueltos con éxito con la soldadura robótica. (Fateh, Farahani, & Khatamianfar, 2010)

Sin embargo para garantizar un aumento en la productividad el rendimiento de los manipuladores debe ser maximizado. Esto puede ser realizado mediante una adecuada planeación de la trayectoria del robot, utilizando técnicas de optimización de alguna función objetivo o de sus parámetros. (Lanzutti, 2011)

Entre estos parámetros o criterios destacan la disminución del tiempo entre recorridos, minimización de desplazamientos en eslabones y articulaciones, aumento de la manipulabilidad (reducción de singularidades), reducción de los requerimientos consumo de energía mecánica o eléctrica. Cada uno de estos ha sido objeto de estudio y han dado lugar a desarrollo de investigaciones que plantean diversos métodos de optimización.

Muchas de las aplicaciones de los robots industriales son basadas en procesos repetitivos donde el ciclo mínimo de tiempo es un factor importante para reducir el tiempo de producción e incrementar la productividad. Entre las investigaciones que han abordado este tema se encuentran:

La investigación desarrollada por (Liao et al., 2010) en la cual se plantea un método que tiene como finalidad optimizar el intervalo de tiempo entre cada sección de la trayectoria usando un algoritmo genético adaptativo (AGA), por medio del cual las trayectorias de los movimientos de las articulaciones en manipuladores de 6 grados de libertad son optimizadas tomando el tiempo mínimo entre dos puntos como índice de rendimiento.

(Kamrani, Berbyuk, & Wäppling, 2009) Diseñan un enfoque resultado de la translación o rotación de la posición del camino de manera independiente respecto a la base del manipulador, el cual en conjunción con el método de superficie de respuesta permite minimizar el tiempo de ciclo de la tarea realizada.

(Zhang & Qi, 2008) Desarrollan una metodología mediante la cual, la posición de múltiples tareas de un robot dentro de la celda de trabajo es optimizada en términos de minimización del tiempo de ciclo. Para aumentar la eficiencia en tiempo el método es descompuesto en 3 etapas. 1) Ubicación individual de tareas: cada tarea es ubicada dentro de una región preferencial y luego rotada sobre su propio marco para obtener la mejor posición. 2) Clasificación de la tarea: se hace para encontrar la mejor secuencia de tareas provista con una secuencia lógica de visita. 3) Optimización total: finalmente la posición de todas las tareas es ajustada simultáneamente en un gran espacio reducido de búsqueda por medio del Método de Recocido Simulado (Simulated Annealing - SA).

Sin embargo el criterio del tiempo mínimo no es aplicable si un camino suave para el movimiento es requerido a velocidades constantes, como es el caso de la soldadura robótica. Cuando los actuadores corren a altas velocidades, en algunas ocasiones causan vibraciones físicas y sacudidas no deseadas en el sistema. Estas vibraciones no deseadas pueden resultar en una amplia variedad de problemas, incluyendo la pérdida de precisión, aumento de consumo de energía y disminución de la vida del actuador (Ayten, Iravani, & Sahinkaya, 2011).

Por otra parte, se identifican una diversidad de técnicas enfocadas en la minimización del desplazamiento en eslabones y articulaciones, entre ellas se tienen:

La investigación realizada por (González, Blanco, & Moreno, 2009), en la cual se emplean algoritmos evolutivos, llamados Evolución Diferencial (Differential Evolution DE), para resolver el problema de la generación óptima de caminos de robots manipuladores no redundantes. Un camino óptimo de la articulación es obtenido usando un esquema variable de Evolución Diferencial tal que el error de posición del efector final y el desplazamiento total de las articulaciones sean minimizadas, previniendo que las condiciones iniciales sean cambiadas. A diferencia de otros enfoques similares, este esquema no solo considera el error de posición del efector final dentro del criterio de optimización sino también el error en la orientación del efector final

(Albert et al., 2009) Proponen un método de planeación de camino usando algoritmos genéticos el cual busca la configuración del manipulador de tres grados de libertad para optimizar los ángulos de las articulaciones. El problema consiste en localizar un camino específico que requiera la menor

cantidad de movimiento angular entre diversos caminos posibles mediante un algoritmo genético que usa un muestreo estocástico.

En las investigaciones mencionadas se evidencia que los métodos desarrollados emplean un único criterio de optimización, lo cual la puede generar comportamientos negativos en los demás parámetros cuando deben ser implementados en la ejecución de tareas específicas.

También es posible encontrar una variedad de investigaciones que han desarrollado métodos multiobjetivo enfocados a manipuladores seriales, tales como:

(Dos Santos, Steffen Jr, & Pereira Saramago, 2006) Presentan una estrategia en la cual el consumo de potencia mecánica y la evasión de obstáculos son consideradas a la vez como metas a ser optimizadas, y el espacio de trabajo es supuesto de ser un ambiente variable en el tiempo debido a la existencia de obstáculos móviles. El método propuesto usa la velocidad como el vector de control y la posición asociada de la articulación como el vector de estado en conjunto con una teoría de optimización de multicriterio. De las teorías de optimización ensayadas por los autores la solución más óptima fue obtenida usando el Método de Programación Lineal Secuencial (Sequential Linear Programming - SLP).

(Dos Santos, Steffen Jr, & Pereira Saramago, 2010) Proponen un enfoque para aumentar la manipulabilidad mientras se disminuye los requerimientos mecánicos de potencia usando técnicas de optimización en manipuladores seriales. El proceso de optimización aplica matrices de transformación translacional y rotacional respecto a un marco de referencia. Como resultado una secuencia óptima de posiciones cinemáticas del robot es determinada. El problema de optimización está basado en el algoritmo de túnel el cual es una metodología heurística diseñada para encontrar el mínimo global de una función, mejorando la robustez del proceso respecto a las estimaciones iniciales.

Las dos investigaciones anteriores poseen metodologías que se basan en optimizaciones multicriterio y tienen como parámetro común la reducción en requerimientos de potencia mecánica. Sin embargo al optimizar mediante este parámetro se puede hacer una subestimación de los requerimientos de energía en el caso donde los actuadores experimentan torques con velocidades iguales a cero. Tampoco considera la pérdida de energía resistiva en el bobinado del motor, como también la pérdida de energía debido a las variaciones en las velocidades del actuador. Estas variaciones afectan los requerimientos de corriente y por lo tanto inducen fuerzas electromotrices en los actuadores (Ur-Rehman et al., 2010).

Al usar la minimización de energía eléctrica como parámetro de optimización se pueden tener en cuenta las pérdidas y gastos de energía que no han sido consideradas en las investigaciones realizadas con los criterios de optimización mencionados anteriormente, razón por la cual este parámetro ha sido foco de investigaciones recientes en la optimización de la planeación de trayectorias en manipuladores. Algunas de estas investigaciones que han usado este parámetro de optimización en manipuladores seriales son:

La realizada por (Sharma & Kaur, 2011), en la cual desarrollan un algoritmo genético para robots de 3 grados de libertad buscando optimizar la planeación de la trayectoria punto a punto. La función objetivo para el algoritmo propuesto es minimizar la energía consumida por cada motor del brazo robot, mientras no exceda un torque máximo predefinido sin colisiones con algún obstáculo en el espacio de trabajo del robot, generando el desplazamiento angular óptimo en el brazo robótico. Se divide en 10 partes la trayectoria definida por punto inicial y punto final para encontrar el ángulo adecuado de la cinemática inversa. El algoritmo genético es usado para calcular los ángulos de las articulaciones para los 10 puntos intermedios y luego la menor distancia posible.

En el enfoque propuesto por (Meike & Ribickis, 2011), se presenta un método de optimización de trayectorias de robot manipuladores para los tipos de movimientos punto a punto. Este enfoque modifica los perfiles de los ejes cinemáticos reduciendo las fases de aceleración, torques en el motor y por ende el total de energía consumida representada en el no incremento del tiempo de ciclo. El método está basado en la suavización del perfil de movimiento asincrónico usando interpolación cúbica B-spline. Los ahorros de energía logrados son inversamente proporcionales a las desviaciones cartesianas, por lo tanto las trayectorias principales del robot o sus fragmentos pueden ser optimizadas solamente donde la ruta entre los puntos del proceso no sean muy relevantes.

Estas investigaciones utilizan la minimización de energía como parámetro de optimización, sin embargo sus métodos están basados en un solo criterio. Esto conlleva a que otros parámetros que son importantes en la planeación de trayectorias en tareas de soldadura puedan ser afectados de manera negativa, como son la posición y orientación final de la herramienta, debido a que las trayectorias optimizadas según parámetro de energía demandarán menor consumo, y las orientaciones del efector final necesarias para garantizar un buen cordón a lo largo de la trayectoria no necesariamente se adaptan a esa condición.

Lo expuesto anteriormente resalta la importancia de contar con métodos multicriterio de optimización de planeación de trayectorias para manipuladores en los cuales se involucre la minimización de la energía eléctrica, por ser un parámetro importante cuando se requiere optimizar las pérdidas de energía en conjunción con otro parámetro buscando que no afecte el desempeño de la tarea realizada.

La manipulabilidad es un criterio adecuado para combinar con el parámetro de minimización de energía en la optimización de la planeación de trayectorias en tareas de soldadura, porque permite cuantificar la cercanía del robot a una singularidad, situación que resulta útil a la hora de reubicar las articulaciones del robot maximizando su destreza. Este criterio fue introducido por (Yoshikawa, 1985) y ha sido la base para diversas investigaciones que van desde mejoras en el diseño cinemático de los manipuladores (Altuzarra, Sandru, Macho, & Amezua, 2010) hasta optimizaciones en la planeación de la trayectoria (Dos Santos, Steffen Jr, & Pereira Saramago, 2010).

En el campo de los manipuladores paralelos se han desarrollado metodologías multiobjetivo que incluyen el parámetro de minimización de energía. A continuación se enuncian algunas investigaciones en las que se evidencia el uso de esta metodología.

(Ur-Rehman et al., 2010) Proponen un método de optimización de trayectorias cuyos objetivos son minimizar la energía consumida por los actuadores, minimizar las fuerzas o momentos de sacudida y minimizar los torques máximos en los actuadores en máquinas cinemáticas paralelas de tres grados de libertad mediante un Algoritmo Genético Multi Objetivo (Multi-Objective Genetic Algorithm - MOGA).

(Khoukhi, Baron, & Balazinski, 2009) Introducen un sistema de planeación de trayectorias el cual minimiza la energía cinética y eléctrica, el tiempo de viaje del robot, separándolo en dos periodos de muestreo y maximizando la medida de la manipulabilidad para evitar singularidades en máquinas cinemáticas paralelas, por medio de una técnica Lagrangiana Aumentada.

Estos métodos, como ya se había mencionado, fueron desarrollados para máquinas cinemáticas paralelas, sin embargo la mayoría de los manipuladores usados en tareas de soldadura de arco son seriales por ser más comunes y asequibles comercialmente. Esto se debe principalmente a la complejidad del diseño, manufactura y calibración de los robots paralelos (Briot & Bonev, 2007).

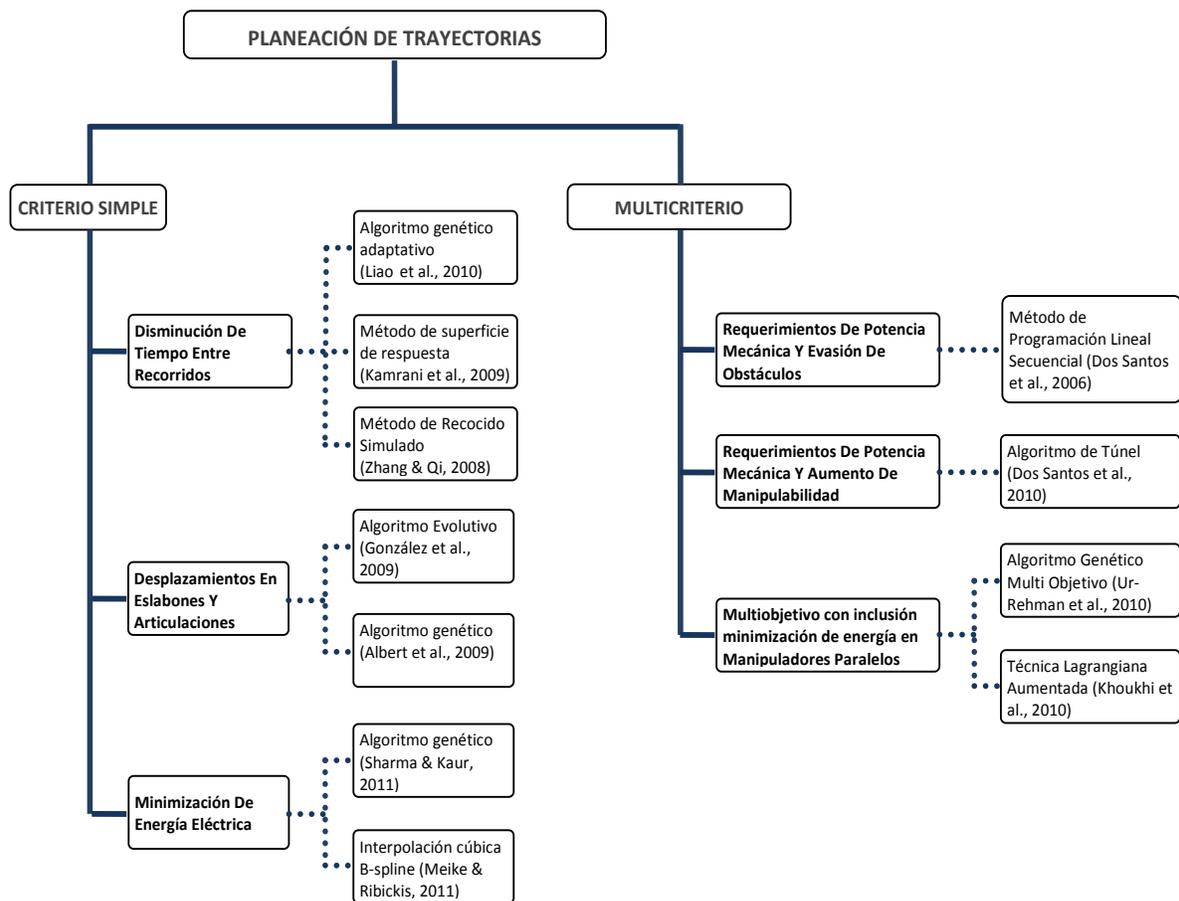


Figura 1 Mapa Conceptual

SÍNTESIS DEL PROBLEMA

A partir de la revisión del estado del arte planteada se detectó que muchos de los métodos de planeación de trayectoria propuestos para manipuladores se enfocaban a la minimización de tiempo y minimización de desplazamiento de las articulaciones y pocos a la minimización de consumo de energía eléctrica. De las investigaciones que tomaron el criterio de minimización del consumo de energía eléctrica, aquellas que lo abordaron como parte de un método multiobjetivo lo realizaron en manipuladores paralelos mediante métodos heurísticos tales como los Algoritmos Genéticos y algoritmo Lagrangiano Aumentado, y no se encontró evidencia de desarrollos en planeación de trayectoria realizados con el Algoritmo Heurístico de Kalman (Heuristic Kalman Algorithm – HKA), el cual fue propuesto por (Toscano & Lyonnet, 2009) cuya principal ventaja es la pequeña cantidad de parámetros de diseño que deban ser definidos por el usuario.

En el proceso de soldadura por arco intervienen una serie de parámetros siendo los más relevantes la corriente, el voltaje y la velocidad de aplicación de la soldadura debido a que afectan la geometría de la soldadura. La velocidad de la soldadura afecta directamente la altura del cordón y la altura de refuerzo (NA et al., 2011), razón por lo cual garantizar una velocidad constante en la aplicación del cordón es lo ideal para lograr una soldadura de óptima calidad.

De los planteamientos anteriores se detectó una oportunidad de investigación en la planeación de trayectorias para manipuladores seriales de seis grados de libertad con muñeca esférica aplicados a tareas de soldadura de arco, por esta razón en el presente trabajo se propone un método de optimización multi-objetivo que integra la minimización del consumo de energía eléctrica y aumento de manipulabilidad basado en el Algoritmo Heurístico de Kalman, idóneo para una pequeña cantidad de parámetros de diseño definidos por el usuario. Al minimizar el consumo de energía eléctrica y combinarlo con un aumento en la manipulabilidad se generará una disminución en el gasto energético del proceso, al igual que trayectorias suaves a velocidades constantes en el efector final, debido a que las articulaciones del manipulador estarán alejadas de posibles singularidades, maximizando su destreza y disminuyendo el esfuerzo soportado en éstas.

OBJETIVO GENERAL

Proponer un método multi-objetivo de planeación de trayectorias por medio de la optimización del parámetro de minimización de energía eléctrica integrado con la manipulabilidad para manipuladores seriales de seis grados de libertad usados en tareas de soldadura de arco en lámina a través del Algoritmo Heurístico de Kalman.

OBJETIVOS ESPECÍFICOS

Modelar matemáticamente el consumo eléctrico y manipulabilidad de manipuladores seriales de seis grados de libertad usados en tareas de soldadura de arco en lámina para proponer funciones objetivo que permitan la optimización de los criterios.

Diseñar un algoritmo multi-objetivo basado en el Algoritmo Heurístico de Kalman, que permita integrar los criterios de minimización de consumo eléctrico y maximización de manipulabilidad.

Validar método de optimización mediante comparación de parámetros de trayectorias optimizadas y trayectorias previamente generadas en manipulador serial de seis grados de libertad usado en tareas de soldadura de arco en lámina.

ORGANIZACIÓN DEL DOCUMENTO

El presente trabajo se encuentra organizado de la siguiente forma:

En el primer capítulo se hace una exposición del marco teórico necesario para el desarrollo del proyecto compuesto por temas de cinemática directa, cinemática inversa, cinemática diferencial, dinámica inversa, algoritmos de optimización y productividad.

El capítulo dos abarca el desarrollo y planteamiento de los modelos de cálculo necesarios para desarrollar el proyecto enfocados en manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica.

En el capítulo tres, se describen los aspectos físicos de los manipuladores PUMA 560 y KUKA KR5 HW ARC al igual que de cinco trayectorias. Partiendo de estos aspectos se exponen los resultados de la optimización de las trayectorias en cada manipulador.

1 MARCO TEÓRICO

A continuación se hace una descripción teórica de la cinemática directa, cinemática inversa, cinemática diferencial, dinámica inversa y algoritmo heurístico de Kalman, los cuales son temas necesarios para poder desarrollar el método de planeación de trayectorias propuesto.

1.1 GENERALIDADES DE LOS MANIPULADORES SERIALES

Un manipulador robótico, es un sistema mecánico comandado mediante un control computarizado que tiene como objetivo ayudar al hombre a realizar tareas manuales. Su principal ventaja radica en su control, debido a que este le permite almacenar y reproducir las veces necesarias una tarea que haya sido programada previamente.

La estructura mecánica de un robot manipulador serial o de cadena abierta consiste en una secuencia de eslabones interconectados por medio de articulaciones. Un manipulador generalmente está compuesto por un brazo que le permite movilidad, una muñeca que le otorga destreza, y un efector final que realiza la tarea requerida por el robot. La tarea del brazo es la de posicionar la muñeca, la cual a su vez es la encargada de orientar el efector final.

Desde el punto de vista topológico que la estructura del manipulador se considere como una cadena cinemática abierta hace referencia a que solo existe una secuencia de eslabones conectándose en sus extremos con la cadena.

La movilidad de un manipulador se asegura por la presencia de articulaciones. La articulación entre dos eslabones consecutivos puede ser realizada ya sea mediante una articulación prismática (P) o una de revolución (R). En una cadena cinemática abierta cada articulación le provee a la estructura un grado de libertad (GDL). Una articulación prismática crea un movimiento translacional relativo entre dos eslabones, en cambio una articulación de revolución crea un movimiento rotacional relativo entre los dos eslabones. Las articulaciones de revolución son preferidas debido que son compactas y confiables frente a las articulaciones prismáticas.

El espacio de trabajo representa la porción del ambiente al cual el manipulador tiene acceso. Su forma y volumen dependen de la estructura del manipulador y de la presencia de límites mecánicos en sus articulaciones (Angeles, 2003).

1.1.1 CLASIFICACIÓN CINEMÁTICA DE LOS MANIPULADORES SERIALES

Los manipuladores seriales industriales disponibles a nivel comercial, generalmente están compuestos por seis o menor cantidad de grados de libertad. Estos manipuladores pueden clasificarse cinemáticamente basándose en las tres primeras articulaciones que conforman el

brazo en los siguientes grupos: Antropomórficos (RRR), Esféricos (RRP), SCARA (RRP), Cilíndricos (RPP), o cartesianos (PPP).

Los manipuladores articulados son también conocidos como de revolución o antropomórficos. Están compuestos por tres articulaciones de revolución (RRR). El eje de revolución de la primera articulación es ortogonal a los ejes de las otras dos los cuales son paralelos (Fig. 2). Su espacio de trabajo es aproximadamente una porción de esfera y su volumen es grande en comparación con la envergadura del robot. Estos se caracterizan por realizar movimientos con altos grados de libertad en espacios reducidos, razón por la cual son los más usados comercialmente (Siciliano et al., 2009).

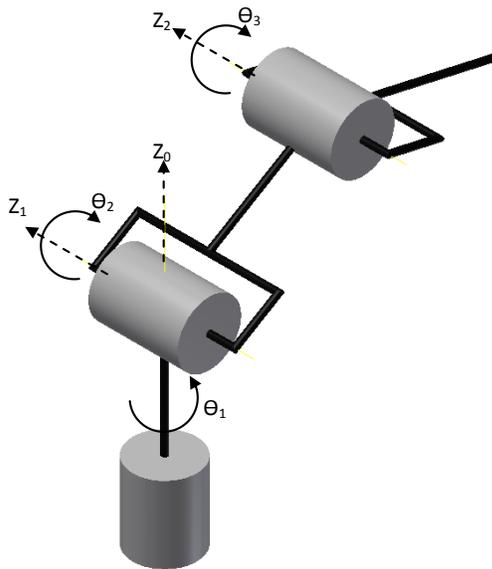


Figura 2 Manipulador Antropomórfico

Reemplazando la tercera articulación en un manipulador de revolución por una articulación prismática se obtiene un manipulador esférico (RRP) (Fig. 3). Este se caracteriza porque mediante coordenadas esféricas se define la posición de su efector final respecto a un Marco de referencia en el cual su origen concuerda con la intersección de los tres ejes z de sus articulaciones.

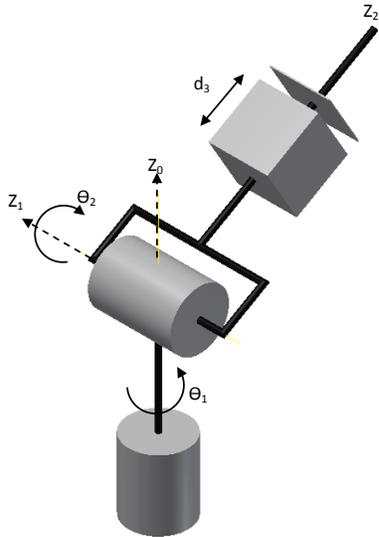


Figura 3 Manipulador Esférico

El brazo SCARA (Selective Compliant Articulated Robot for Assembly) es un manipulador popular usado principalmente en operaciones de ensamble. Aunque su estructura es tipo RRP, es muy diferente al manipulador esférico tanto en apariencia como en rango de aplicación. A diferencia del manipulador esférico que tiene el eje z_0 de la primera articulación, perpendicular a los ejes z_1 y z_2 de las otras dos articulaciones, el SCARA posee todos los ejes z de sus articulaciones paralelos entre sí (Fig. 4).

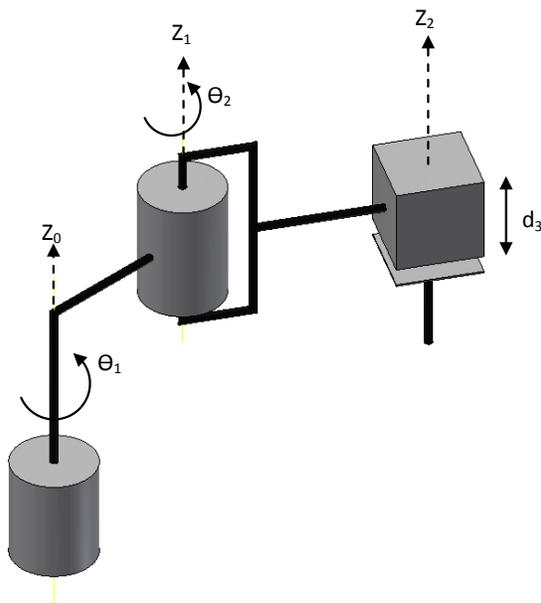


Figura 4 Manipulador SCARA

En el manipulador cilíndrico la primera articulación es de revolución y produce una rotación sobre la base, mientras que la segunda y tercera articulación, son prismáticas (RPP) (Fig. 5). Se caracteriza por poseer un espacio de trabajo definido en coordenadas cilíndricas respecto a su base.

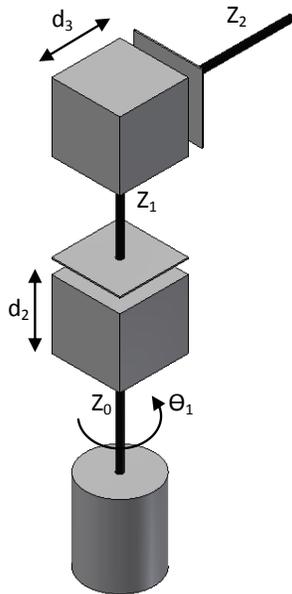


Figura 5 Manipulador Cilíndrico

Un manipulador en el cual sus primeras tres articulaciones son prismáticas es conocido como manipulador cartesiano (Fig. 6). Su espacio de trabajo está definido en coordenadas cartesianas respecto a su base (Spong, Hutchinson, & Vidyasagar, Robot Modeling and Control, 2005).

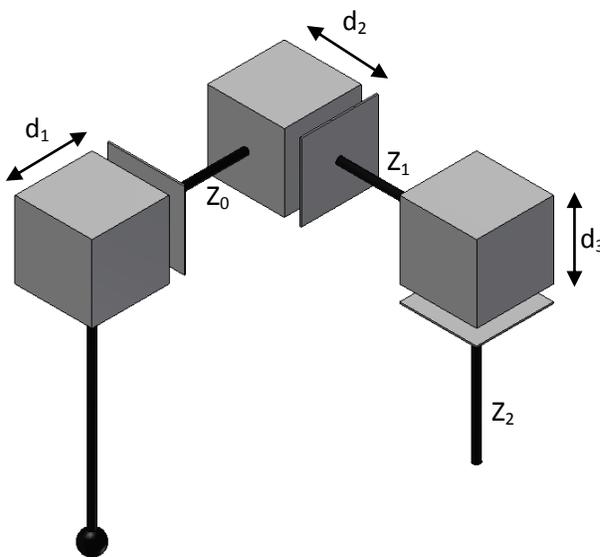


Figura 6 Manipulador Cartesiano

1.1.2 MUÑECA Y EFECTOR FINAL

La muñeca de un manipulador hace referencia a las articulaciones en la cadena cinemática entre el brazo y el efector final. Esta es la encargada de dar la orientación del efector final, razón por la cual los grados de libertad de orientación de este dependen directamente de los grados de libertad de la muñeca.

La configuración más común de muñecas consiste de tres articulaciones de revolución. La mayoría de los manipuladores están diseñados con muñecas esféricas, lo cual quiere decir que los tres ejes de las articulaciones de la muñeca se intersecan en un punto común (Fig. 7). La muñeca esférica simplifica considerablemente el análisis cinemático, porque permite de manera efectiva, lograr un desacople en la posición y orientación del efector final.

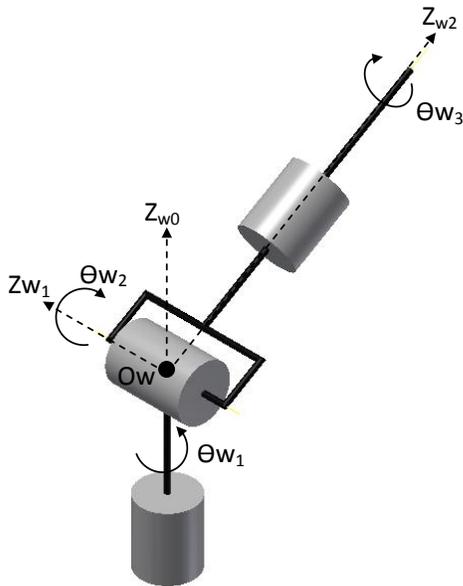


Figura 7 Muñeca esférica

El ensamble del brazo y la muñeca de un robot son usados primordialmente para posicionar el efector final y cualquier herramienta que deba cargar. Es la herramienta o efector final la que en realidad ejecuta el trabajo. El tipo de efector final más simple son las pinzas, que por lo general son capaces de efectuar sólo dos acciones, la apertura y el cierre, siendo adecuadas para la transferencia de materiales, pero no son adecuadas para otras tareas más complejas, como la soldadura, pintura, rectificado, etc., para las cuales se debe contar con un efector final especializado (Spong, Hutchinson, & Vidyasagar, 2005).

1.2 CINEMÁTICA MANIPULADORES SERIALES

La cinemática es una ciencia que estudia el movimiento sin considerar las fuerzas que lo generan. Dentro de las ramas de la cinemática, destacan aquellas que profundizan en la posición, velocidad, aceleración, y las derivadas de orden superior de las variables de posición (con respecto al tiempo o cualquier otra variable). El estudio de la cinemática de los manipuladores hace referencia a todas sus propiedades geométricas y de movimiento basadas en el tiempo.

El número de grados de libertad que un manipulador posee, hace referencia al número de variables de posición independientes que tendrían que ser especificadas con el fin de localizar todas las partes de su mecanismo. En el caso de los manipuladores industriales seriales, cada posición de la articulación es usualmente definida con una sola variable, y el número de articulaciones es igual al número de grados de libertad.

1.2.1 CINEMÁTICA DIRECTA MANIPULADORES SERIALES

El problema cinemático directo hace referencia a la relación entre las articulaciones individuales de un robot manipulador y la posición y orientación de la herramienta o efector final. Las variables de articulación son los ángulos comprendidos entre los eslabones, para el caso de articulaciones rotacionales, y de la extensión de los eslabones para el caso de articulaciones prismáticas.

Una convención usada comúnmente para seleccionar los marcos de referencia para cada articulación en manipuladores robóticos es la convención Denavit-Hartenberg, o convención DH. En esta convención la matriz de transformación homogénea $i-1A_i$ entre dos eslabones consecutivos y articulación común i , es representada como un producto de las cuatro transformaciones básicas:

Matriz de rotación de un ángulo θ_i alrededor del eje Z_{i-1} :

$$T(z, \theta_i) = \begin{bmatrix} \cos(\theta_i) & -\text{Sen}(\theta_i) & 0 & 0 \\ \text{Sen}(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Matriz de traslación constante de una distancia d_i sobre el eje Z_{i-1} :

$$T(0,0, d_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Matriz de traslación constante de una distancia a_i sobre el eje X_i :

$$T(0,0,a_i) = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Matriz de rotación de un ángulo α_i alrededor del eje X_i :

$$T(x,\alpha_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Siendo A_i :

$${}^{i-1}A_i = T(z,\theta_i)T(0,0,d_i)T(a_i,0,0)T(x,\alpha_i) \quad (5)$$

$${}^{i-1}A_i = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\cos(\theta_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Los cuatro parámetros a_i , α_i , d_i , y θ_i , son conocidos como longitud del eslabón, orientación del eslabón, separación del eslabón, y ángulo de articulación, respectivamente. Para un eslabón específico los parámetros, θ_i para articulaciones de revolución y d_i para articulaciones prismáticas, son variables. Los parámetros restantes son constantes.

Para obtener el modelo cinemático de un manipulador según la convención DH (Fig. 8), se deben seguir los siguientes pasos:

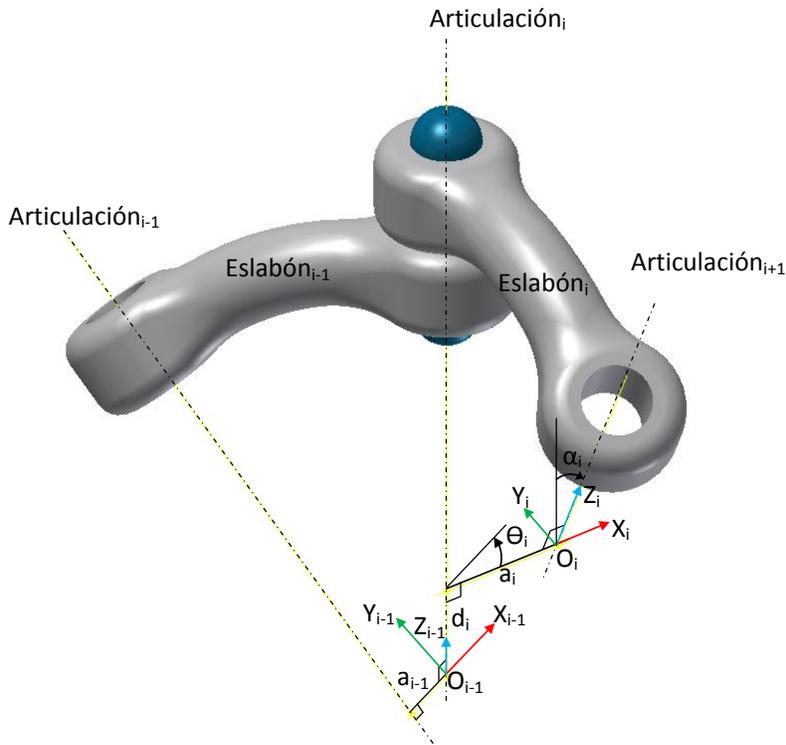


Figura 8 Modelo cinemático general eslabones manipulador según convención DH

Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y finalizando con n (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot.

Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad) y finalizando en n.

Identificar el eje de cada articulación. Si ésta es rotativa, el eje coincide con su eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.

Para i de 0 a $n-1$ situar el eje Z_i sobre el eje de la articulación $i+1$.

Ubicar el origen del sistema de coordenadas la base $\{O_0\}$ en cualquier punto del eje Z_0 . Los ejes X_0 , Y_0 se situarán de modo que formen un sistema dextrógiro con Z_0 .

Para i de 1 a $n-1$, ubicar el sistema de coordenadas $\{O_i\}$ (solidario al eslabón i) en la intersección del eje Z_i con la línea normal común a Z_{i-1} y Z_i . Si ambos ejes se intersecan se ubicaría $\{O_i\}$ en el punto de corte. Si fuesen paralelos $\{O_i\}$ se ubicaría en la articulación $i+1$

Ubicar X_i en la línea normal común a Z_{i-1} y Z_i .

Ubicar Y_i de modo que forme un sistema dextrógiro con X_i y Z_i .

Situar el sistema $\{O_n\}$ en el extremo del robot de modo que Z_n coincida con la dirección de Z_{n-1} y X_n sea normal a Z_{n-1} y Z_n .

Definir θ_i como el ángulo que se debe girar en torno a Z_{i-1} para que X_{i-1} y X_i sean paralelos.

Obtener d_i como la distancia, medida a lo largo de Z_{i-1} , que se debe desplazar $\{O_{i-1}\}$ buscando que X_i y X_{i-1} queden alineados.

Obtener a_i como la distancia medida a lo largo de X_i (ahora coincidente con X_{i-1}) que se debe desplazar el nuevo $\{O_{i-1}\}$ para que su origen coincida con $\{O_i\}$.

Obtener α_i como el ángulo que se debe girar en torno a X_i (ahora con X_{i-1}), para que el nuevo $\{S_{i-1}\}$ coincida totalmente con $\{O_i\}$.

Hallar las matrices de transformación homogénea A_i .

Hallar matriz de transformación homogénea T entre la base y el extremo del robot. Con esta se define la orientación y posición del efector final referido a la base en función de las n -coordenadas articulares.

$$T = {}^0A_1 {}^1A_2 \dots {}^{n-1}A_n \quad (7)$$

$$T = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

Siendo:

n : Vector normal del extremo libre del manipulador

s : Vector de deslizamiento del extremo libre del manipulador.

a : Vector de aproximación

p : Vector de posición del extremo libre del manipulador.

La matriz T puede ser descompuesta en una matriz R , la cual contiene la información de la orientación del efector final, y el vector P que contiene información de la posición del efector final. El anterior supuesto es útil cuando se requiere hacer cálculos solo de las orientaciones entre las articulaciones (Spong, Hutchinson, & Vidyasagar, Robot Modeling and Control, 2005).

$$R = \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} \quad (9)$$

$$P = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (10)$$

Quedando:

$$T = \begin{bmatrix} R(\theta) & P(\theta) \\ 0^T & 1 \end{bmatrix} \quad (11)$$

1.2.2 CINEMÁTICA INVERSA MANIPULADORES SERIALES

Por medio de la cinemática inversa se busca encontrar los valores de las variables de articulación en términos de la posición y orientación del efector final. Encontrar la solución a la cinemática inversa de un manipulador puede ser una tarea algo compleja debido a que generalmente está descrita mediante ecuaciones no lineales. Soluciones de forma cerrada pueden no ser posibles y múltiples, infinitas o soluciones imposibles pueden llegar a presentarse.

Sin embargo algunos casos especiales tienen una solución de forma cerrada y pueden ser resueltos. Entre estos se encuentran los manipuladores seriales de seis grados de libertad con muñeca esférica (Fig. 9), en los cuales es posible desacoplar el problema de la cinemática inversa en dos soluciones más simples, conocidas respectivamente como, cinemática inversa de posición, y cinemática inversa de orientación. El desacople cinemático en manipuladores de muñeca esférica es posible gracias a que los tres ejes de revolución de la muñeca se intersecan en un punto común (O_w). (Pires, 2007)

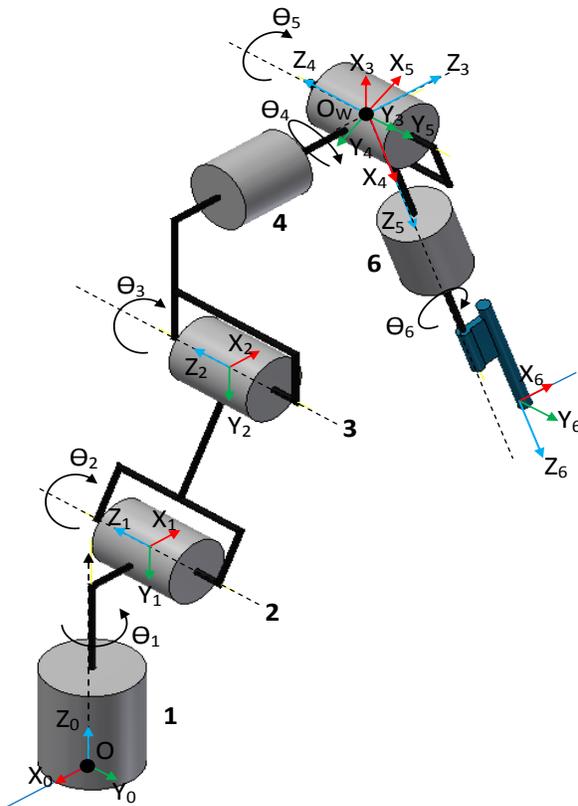


Figura 9 Manipulador de seis grados de libertad con muñeca esférica.

1.3 CINEMÁTICA DIFERENCIAL MANIPULADORES SERIALES

El objetivo de la cinemática diferencial es encontrar las relaciones entre las velocidades de las articulaciones y las velocidades lineales y angulares correspondientes al efector final o herramienta. En otras palabras se busca expresar la velocidad lineal del efector final \dot{P} y su velocidad angular ω , como una función de las velocidades de las articulaciones $\dot{\theta}$.

Partiendo de:

$$\omega = J_o(\theta)\dot{\theta} \quad (12)$$

$$\dot{P} = J_p(\theta)\dot{\theta} \quad (13)$$

Donde J_p es una matriz (3xn) que relaciona la contribución de las velocidades de las articulaciones $\dot{\theta}$ con el vector de las velocidades lineales \dot{P} del efector final. Y J_o es una matriz (3xn) que relaciona la contribución de las velocidades de las articulaciones $\dot{\theta}$ con el vector de las velocidades angulares ω del efector final. De una forma más compacta las ecuaciones anteriores pueden ser escritas como:

$$v = \begin{bmatrix} \omega \\ \dot{P} \end{bmatrix} = J(\theta)\dot{\theta} \quad (14)$$

Ecuación que representa la cinemática diferencial del manipulador, y donde:

$$J = \begin{bmatrix} J_o \\ J_p \end{bmatrix} \quad (15)$$

Es una matriz (6xn) que representa la Jacobiana geométrica del manipulador, que en general es una función de las variables de articulación.

La relación entre las aceleraciones articulares y las aceleraciones lineales del efector final o herramienta, puede obtenerse realizando la derivada respecto al tiempo de la ecuación (14):

$$\dot{v} = \begin{bmatrix} \dot{\omega} \\ \ddot{P} \end{bmatrix} = \dot{J}(\theta)\dot{\theta} + J(\theta)\ddot{\theta} \quad (16)$$

1.3.1 CALCULO JACOBIANA GEOMÉTRICA

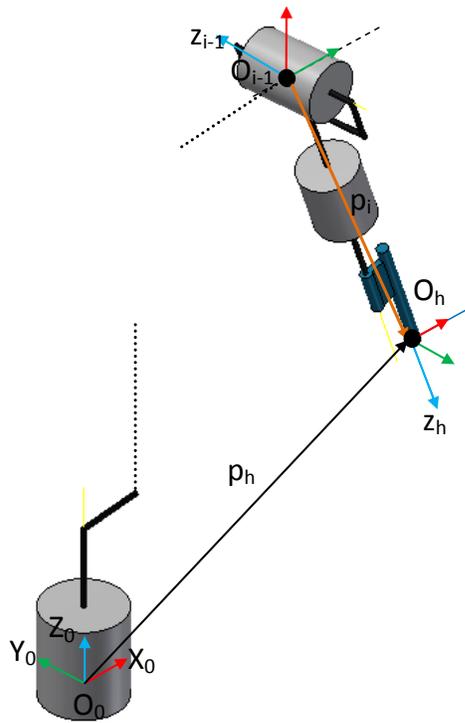


Figura 10 Vectores necesarios para calcular la contribución de la velocidad de una articulación en la velocidad del efector final

Para proceder con el cálculo de la Jacobiana, es conveniente separar la velocidad lineal y la velocidad angular (Siciliano et al., 2009).

En la contribución a la velocidad lineal, la derivada respecto al tiempo del vector de posición $P(\Theta)$ (Fig. 10) puede escribirse como:

$$\dot{P} = \sum_{i=1}^n \frac{\partial p_h}{\partial \theta_i} \dot{\theta}_i = \sum_{i=1}^n J_{p_i} \dot{\theta}_i \quad (17)$$

Esta expresión muestra como el vector de velocidades lineales \dot{P} , puede ser obtenido como la suma de los términos $J_{p_i} \dot{\theta}_i$. Cada término representa la contribución de la velocidad de una articulación i a la velocidad lineal del efector final.

La contribución a la velocidad lineal puede ser calculada con referencia al origen del Marco del efector final, así:

$$J_{p_i} \dot{\theta}_i = \omega_{i-1,i} \times p_i = \dot{\theta}_i z_{i-1} \times p_i \quad (18)$$

Por consiguiente:

$$J_{P_i} = z_{i-1} \times p_i \quad (19)$$

Para la contribución a la velocidad angular del efector final, se tiene que:

$$\omega = \sum_{i=1}^n \omega_{i-1,i} = \sum_{i=1}^n J_{O_i} \dot{\theta}_i \quad (20)$$

Dónde:

$$J_{O_i} \dot{\theta}_i = \omega_{i-1,i} = \dot{\theta}_i z_{i-1} \quad (21)$$

Por consiguiente:

$$J_{O_i} = z_{i-1} \quad (22)$$

Resumiendo, la Jacobiana puede ser seccionada en vectores columna (3x1) J_{P_i} y J_{O_i} , así:

$$J = \begin{bmatrix} J_{O1} & \dots & J_{On} \\ J_{P1} & & J_{Pn} \end{bmatrix} \quad (23)$$

1.3.2 CINEMÁTICA INVERSA DIFERENCIAL

La cinemática inversa diferencial representa una aplicación lineal entre las velocidades en el espacio articular y las velocidades en el espacio operacional. Este hecho sugiere la posibilidad de utilizar las ecuaciones de la cinemática inversa diferencial para resolver el problema de cinemática inversa diferencial.

Partiendo de la anterior afirmación y además, si la matriz de la Jacobiana inversa existe en un determinado vector de valores de posiciones articulares Θ_i , las velocidades de las articulaciones pueden ser obtenidas despejando la ecuación (14), obteniendo el siguiente resultado:

$$\dot{\theta} = J^{-1}(\theta) \begin{bmatrix} \omega \\ \dot{P} \end{bmatrix} \quad (24)$$

Usando la ecuación anterior ecuación (24) la ecuación de aceleración del efector final (16) puede ser expresada de la siguiente forma:

$$\begin{bmatrix} \dot{\omega} \\ \ddot{P} \end{bmatrix} = \dot{J}(\theta) J^{-1}(\theta) \begin{bmatrix} \omega \\ \dot{P} \end{bmatrix} + J(\theta) \ddot{\theta} \quad (25)$$

Partiendo de la expresión anterior, la ecuación de las aceleraciones articulares queda finalmente expresada como:

$$\ddot{\theta} = J^{-1}(\theta)J(\theta)\begin{bmatrix} \dot{\omega} \\ \dot{P} \end{bmatrix} - J^{-1}(\theta)\dot{J}(\theta)J^{-1}(\theta)\begin{bmatrix} \omega \\ \dot{P} \end{bmatrix} \quad (26)$$

1.3.3 MANIPULABILIDAD TRANSLACIONAL Y ROTACIONAL MANIPULADORES SERIALES

La eficiencia con la cual los desplazamientos articulares de un robot se convierten en movimientos de su órgano terminal es un criterio usado para medir el desempeño, y se conoce como índice de manipulabilidad. La medida de desempeño juega un papel importante en el diseño y aplicabilidad de un manipulador para una tarea específica, gracias a que esta permite identificar trayectorias libres de configuraciones singulares

Uno de los métodos más comunes encontrados en la literatura para medir la manipulabilidad es el propuesto por (Yoshikawa, 1987) y ha sido usado como base para el desarrollo de trabajos que se dedican al estudio del desempeño de brazos manipuladores, manipuladores paralelos, e incluso es aplicado a la biomecánica del brazo humano. El índice de manipulabilidad también se usa ampliamente como parámetro a maximizar en trabajos que proponen métodos de optimización de ubicación de la base del robot, y en trabajos que desarrollan métodos para planeación de trayectorias en manipuladores (Ramírez, Jaramillo, & Duque, 2012)

El método para calcular la manipulabilidad translacional y rotacional descrito en (Yoshikawa, 1991) propone descomponer un manipulador de seis grados de libertad con muñeca esférica en dos secciones, una compuesta por las tres primeras articulaciones de rotación, la cual será llamada sección de brazo, y la otra por las tres articulaciones restantes que conforman la sección de la muñeca esférica, con el fin de hallar la manipulabilidad translacional y rotacional del manipulador al igual que sus singularidades.

Para ello se definirá un sistema de coordenadas con origen O_i , siendo $i = 1, 2, \dots, 6$, fijo a cada eslabón, y se nombran las primeras tres articulaciones que conforman la sección del brazo como n_{A1} , n_{A2} , n_{A3} , y las tres correspondientes a la muñeca como n_{W4} , n_{W5} , n_{W6} . Para este caso el efector final coincide con el último eslabón, por lo que el origen O_E es el punto central del efector final. Adicionalmente se aprecia que los ejes de las articulaciones de la sección de la muñeca se intersecan en el origen O_W . Además p_i denota los vectores de O_i (con $i = 1$ a 3) hasta O_h , y p_w representa el vector de O_W a O_h

Para lo cual la matriz Jacobiana del manipulador está dada por:

$$J = \begin{bmatrix} J_{A1} & J_{A2} & J_{A3} & J_{W4} & J_{W5} & J_{W6} \end{bmatrix} \quad (27)$$

Donde para $i = 1, 2, 3$, se tiene que:

$$J_{Ai} = \begin{bmatrix} z_i \times p_i \\ z_i \end{bmatrix} \quad (28)$$

Y para $i = 4, 5, 6$:

$$J_{wi} = \begin{Bmatrix} z_i \times pw \\ z_i \end{Bmatrix} \quad (29)$$

Donde z_i denota el vector unidad en la dirección del eje Z y x denota el producto del vector.

Por lo tanto la matriz Jacobiana queda dada por:

$$J = \begin{bmatrix} z_1 \times p_1 & z_2 \times p_2 & z_3 \times p_3 & z_4 \times p_w & z_5 \times p_w & z_6 \times p_w \\ z_1 & z_2 & z_3 & z_4 & z_5 & z_6 \end{bmatrix} \quad (30)$$

Y la medida de la manipulabilidad (w) es:

$$w = \det \sqrt{JJ^t} \quad (31)$$

Presentando el vector pw que hace parte del producto cruz como su matriz antisimétrica:

$$[p_w \times] = \begin{bmatrix} 0 & -p_{wz} & p_{wy} \\ p_{wz} & 0 & -p_{wx} \\ -p_{wy} & p_{wx} & 0 \end{bmatrix} \quad (32)$$

Y denotando I_3 como matriz identidad de 3 x 3, se tiene que:

$$J = \begin{bmatrix} I_3 & [p_w \times] \\ 0 & I_3 \end{bmatrix} \times \begin{bmatrix} I_3 & 0 \\ [z_1 \ z_2 \ z_3] [z_1 \times p_{1w} \ z_2 \times p_{2w} \ z_3 \times p_{3w}]^{-1} & I_3 \end{bmatrix} \times \begin{bmatrix} z_1 \times p_{1w} & z_2 \times p_{2w} & z_3 \times p_{3w} & 0 & 0 & 0 \\ 0 & 0 & 0 & z_4 & z_5 & z_6 \end{bmatrix} \quad (33)$$

Donde $piw = p_i - pw$, el cual es el vector de O_i a O_h . Por lo tanto:

$$w = \det [z_1 \times p_{1w} \ z_2 \times p_{2w} \ z_3 \times p_{3w}] \times \sqrt{\det \left\{ \begin{bmatrix} z_4 & z_5 & z_6 \end{bmatrix} \begin{bmatrix} z_4 & z_5 & z_6 \end{bmatrix}^T \right\}} \quad (34)$$

Si se deja:

$$w_A = \det [z_1 \times p_{1w} \ z_2 \times p_{2w} \ z_3 \times p_{3w}] \quad (35)$$

Y

$$w_W = \sqrt{\det \left\{ \begin{bmatrix} z_4 & z_5 & z_6 \end{bmatrix} \begin{bmatrix} z_4 & z_5 & z_6 \end{bmatrix}^T \right\}} \quad (36)$$

Se tiene que:

$$w = w_A w_W \quad (37)$$

Si se asume que el punto final de la sección del brazo es O_w , entonces la matriz:

$$J_A = [z_1 \times p_{1w} \ z_2 \times p_{2w} \ z_3 \times p_{3w}] \quad (38)$$

Es la Jacobiana relacionada con el vector de velocidad $\begin{bmatrix} \dot{q}_1 & \dot{q}_2 & \dot{q}_3 \end{bmatrix}^T$, de las articulaciones con la velocidad translacional de este punto final. Por consiguiente w_A puede ser llamada como medida de manipulabilidad translacional de la sección del brazo.

Similarmente, la matriz:

Es la Jacobiana relacionada con el vector de velocidad $\begin{bmatrix} \dot{q}_4 & \dot{q}_5 & \dot{q}_6 \end{bmatrix}^T$, de las articulaciones con la velocidad rotacional del efector final. Por consiguiente w_w , será la medida de manipulabilidad rotacional del efector final o de la sección de muñeca.

1.4 DINÁMICA MANIPULADORES SERIALES

El modelo dinámico de un manipulador es la relación existente entre los torques (y/o fuerzas) aplicadas a los actuadores y las posiciones de las articulaciones, velocidades y aceleraciones. Juega un papel importante para la simulación del movimiento, el análisis de la estructura del manipulador, y diseño de algoritmos de control. El análisis del modelo dinámico puede ser de ayuda para el diseño mecánico de prototipos de brazos. El cálculo de fuerzas y torques proveen información útil de movimiento usada para el diseño de articulaciones, transmisiones y motores del robot (Siciliano, Sciavicco, Villani, & Oriolo, 2009).

El modelo dinámico se obtiene a partir de leyes físicas conocidas tales como la mecánica Newtoniana y la mecánica Lagrangiana. Diversos tipos de formulaciones para hallar la dinámica de los manipuladores han sido desarrolladas: Newton-Euler, Lagrange-Euler, D' Alembert. Sin embargo ellas son equivalentes entre sí, debido a que definen el mismo fenómeno físico, es decir, las dinámicas de los cuerpos rígidos reunidas para definir un robot. Entre estos tipos de formulaciones, la formulación de Newton-Euler es la más compacta y la más eficiente computacionalmente (Pires, 2007).

Existen generalmente dos problemas relacionados con la dinámica del manipulador que han sido objeto de estudio. En el primer problema partiendo de un punto de la trayectoria, las posiciones de las articulaciones, velocidades y aceleraciones, se busca encontrar el vector de torque de las articulaciones. Este planteamiento de la dinámica es conocido como dinámica Inversa y es útil cuando se desea desarrollar un control para el manipulador. El segundo problema calcula como el manipulador se moverá bajo la aplicación de un conjunto de torques en las articulaciones. Es decir dado un vector de torques, se calcula el movimiento resultante del manipulador, las posiciones de las articulaciones, velocidades y aceleraciones. Este planteamiento se conoce como dinámica directa y es útil para simular el manipulador (Craig, 2005)

1.4.1 DINÁMICA INVERSA SEGÚN FORMULACIÓN NEWTON EULER PARA MANIPULADOR SERIAL

La formulación Newton Euler se basa en el balance de todas las fuerzas actuando en los eslabones genéricos del manipulador.

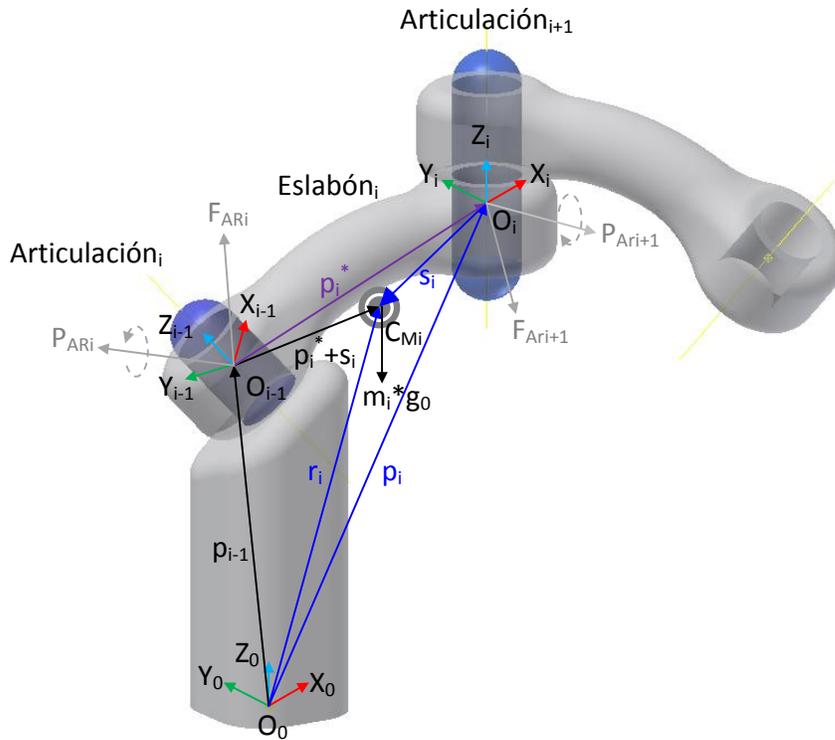


Figura 11 Caracterización del Eslabón para la formulación Newton Euler

Considerando el eslabón genérico i de la cadena cinemática de un manipulador (Fig. 11), y tomando como referencia el centro de masa C_{Mi} de este, se caracterizan los siguientes parámetros:

m_i : Masa del eslabón genérico.

I_i : Tensor de inercia del eslabón genérico.

$p_i^* + s_i$: Vector del origen de Marco de referencia (i-1) al centro de masa C_{Mi}

s_i : Vector del origen de Marco de referencia i al centro de masa C_{Mi}

p_i^* : Vector del origen de Marco de referencia (i-1) al origen del Marco de referencia i .

p_i : Vector del origen de Marco de referencia base (0) al origen del Marco de referencia i .

p_{i-1} : Vector del origen de Marco de referencia base (0) al origen del Marco de referencia $i-1$.

r_i : Vector del origen de Marco de referencia base (0) al centro de masa C_{Mi} .

Las velocidades y aceleraciones a considerar son:

ω_i : Velocidad angular del eslabón.

A_{CMi} : Aceleración lineal del centro de masa C_{Mi}

$\dot{\omega}_i$: Aceleración angular del eslabón.

Las fuerzas y momentos a considerar son:

F_i : Fuerza ejercida por el eslabón i-1 en el eslabón i.

F_{i+1} : Fuerza ejercida por el eslabón i+1 en el eslabón i.

P_{ARi} : Momento ejercido por el eslabón i-1 en el eslabón i. respecto al origen del Marco de referencia i-1.

P_{ARi+1} : Momento ejercido por el eslabón i+1 en el eslabón i. respecto al origen del Marco de referencia i.

Inicialmente, todos los vectores y matrices se asumen expresados con referencia al Marco base.

Con base en estos parámetros se define la ecuación de Newton para el movimiento de translación del centro de masa del eslabón como:

$$F_i - F_{i+1} = m_i A_{CMi} \quad (39)$$

Y se define la ecuación de Euler para el movimiento de rotación del eslabón (referida al momento del centro de masa) como

$$P_{ARi} + F_i \times (p_i^* + s_i) - P_{ARi+1} - F_{i+1} \times p_i^* = \frac{d}{dt}(I_i \omega_i) \quad (40)$$

Cabe anotar que la fuerza gravitacional $m_i g_0$ no genera momento alguno porque está concentrada en el centro de masa.

Desarrollando el lado derecho de la ecuación 40, se tiene que:

$$\frac{d}{dt}(I_i \omega_i) = I_i \dot{\omega}_i + \omega_i \times (I_i \omega_i) \quad (41)$$

Donde el segundo término representa el torque giroscópico inducido por la dependencia de I_i en la orientación del eslabón.

Sustituyendo 41 en 40, la ecuación resultante de Euler es:

$$P_{ARi} + F_i \times (p_i^* + s_i) - P_{ARi+1} - F_{i+1} \times p_i^* = I_i \dot{\omega}_i + \omega_i \times (I_i \omega_i) \quad (42)$$

Por lo tanto, el par/ fuerza generalizado en la articulación i se expresa por:

$$\tau_i = P_{ARi}^T z_{i-1} + b_i \dot{\theta}_i \quad (43)$$

Donde z_{i-1} es un vector unitario igual a $[0 \ 0 \ 1]^T$, y el segundo término del lado derecho de la ecuación involucra la fricción presente en la articulación mediante el coeficiente de rozamiento viscoso b_i , y su velocidad angular (Siciliano, Sciavicco, Villani, & Oriolo, 2009).

1.4.1.1 Aceleraciones De Los Eslabones en el manipulador serial

La aceleración angular y lineal para un eslabón i y para su centro de masa se halla partiendo de las fórmulas básicas de velocidad lineal y angular para un eslabón genérico, las cuales son:

Velocidad angular eslabón i :

$$\omega_i = \omega_{i-1} + \dot{\theta}_i z_{i-1} \quad (44)$$

Velocidad lineal eslabón i :

$$v_i = v_{i-1} + \omega_i \times p_i^* \quad (45)$$

Velocidad lineal centro masa eslabón i :

$$v_i = v_i + \omega_i \times s_i \quad (46)$$

Derivando respecto al tiempo las ecuaciones anteriores, se tiene que:

Aceleración angular eslabón i :

$$\dot{\omega}_i = \dot{\omega}_{i-1} + \ddot{\theta}_i z_{i-1} + \omega_{i-1} \times (\dot{\theta}_i z_{i-1}) \quad (47)$$

Aceleración lineal eslabón i :

$$\dot{v}_i = \dot{v}_{i-1} + \dot{\omega}_i \times p_i^* + \omega_i \times (\omega_i \times p_i^*) \quad (48)$$

Y finalmente la aceleración del centro de masa del eslabón i (Siciliano, Sciavicco, Villani, & Oriolo, 2009):

$$A_{CMi} = \dot{v}_i + \dot{\omega}_i \times s_i + \omega_i \times (\omega_i \times s_i) \quad (49)$$

1.4.1.2 Algoritmo Recursivo

Debe aclararse que el resultado obtenido con las ecuaciones de movimiento de Newton Euler no son de forma cerrada, debido a que el movimiento de un eslabón esta acoplado al movimiento de otro eslabón a través de la relación cinemática de velocidades y aceleraciones

Una vez que las posiciones, las velocidades y las aceleraciones de las articulaciones se conocen, puede procederse a calcular las velocidades y aceleraciones de los eslabones, usando las ecuaciones de Newton-Euler para encontrar las fuerzas y momentos aplicados en el efector final.

Por otro lado, las velocidades y aceleraciones de los eslabones se calculan recursivamente comenzando con la velocidad y aceleración del eslabón base.

En resumen, un algoritmo computacionalmente recursivo puede ser construido de manera que cuente con una recursión hacia delante con respecto a la propagación de las velocidades y aceleraciones y una recursión hacia atrás para la propagación de las fuerzas y momentos a lo largo de la estructura.

Para la recursión hacia delante, una vez que las posiciones, velocidades y aceleraciones de las articulaciones, y las velocidades y articulaciones del eslabón base son especificadas, se procede a calcular las velocidades y aceleraciones angulares, aceleración lineal y aceleración del centro de masa del eslabón analizado.

Una vez se ha calculado las velocidades y aceleraciones con la recursión hacia delante, del eslabón base hacia el efector final, una recursión hacia atrás puede ser realizada para hallar las fuerzas y momentos (Siciliano et al., 2009).

El algoritmo es computacionalmente más eficiente, si todos los vectores se referencian con el Marco actual del eslabón i analizado. Esto implica que todos los vectores deban ser transformados del Marco $i+1$ al Marco i deben ser multiplicados por la matriz de rotación ${}^iR_{i+1}$, mientras que los vectores que deban ser transformados del Marco $i-1$ al Marco i deben ser multiplicados por la matriz de rotación ${}^{i-1}R_i^T$, razón por la cual las ecuaciones de velocidades, articulaciones, fuerzas y pares de los eslabones quedan reescritas como:

Velocidad angular eslabón i :

$${}^i\omega_i = {}^{i-1}R_i^T ({}^{i-1}\omega_{i-1} + \dot{\theta}_i z_0) \quad (50)$$

Aceleración angular eslabón i :

$${}^i\dot{\omega}_i = {}^{i-1}R_i^T \left[{}^{i-1}\dot{\omega}_{i-1} + \ddot{\theta}_i z_0 + {}^{i-1}\omega_{i-1} \times (\dot{\theta}_i z_0) \right] \quad (51)$$

Aceleración lineal eslabón i :

$${}^i\dot{v}_i = {}^{i-1}R_i^T * {}^{i-1}\dot{v}_{i-1} + {}^i\dot{\omega}_i \times {}^i p_i^* + {}^i\omega_i \times ({}^i\omega_i \times {}^i p_i^*) \quad (52)$$

Aceleración del centro de masa del eslabón i :

$${}^i A_{CMi} = {}^i\dot{v}_i + {}^i\dot{\omega}_i \times {}^i s_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i s_i) \quad (53)$$

Fuerza ejercida por el eslabón $i-1$ en el eslabón i :

$${}^i F_i = {}^i R_{i+1} * {}^{i+1} F_{i+1} + m_i {}^i A_{CMi} \quad (54)$$

Momento ejercido por el eslabón $i-1$ en el eslabón i . respecto al origen del Marco de referencia $i-1$.

$${}^i P_{ARi} = {}^i R_{i+1} * {}^{i+1} P_{ARi+1} + {}^i R_{i+1} * {}^{i+1} F_{i+1} \times {}^i p_i^* - {}^i F_i \times ({}^i p_i^* + {}^i s_i) + {}^i I_i {}^i \dot{\omega}_i + {}^i \omega_i \times ({}^i I_i {}^i \omega_i) \quad (55)$$

Par/ fuerza generalizado en la articulación i se expresa por:

$$\tau_i = {}^i R_{i+1} * {}^i P_{ARi}^T z_0 + b_i \dot{\theta}_i \quad (56)$$

Las ecuaciones anteriores poseen una gran ventaja debido a que las cantidades ${}^i I_i, {}^i S_i$, son constantes y además z_0 es igual a $[0 \ 0 \ 1]^T$.

1.4.1.3 Inclusión de la gravedad en el Algoritmo Recursivo

El efecto de la gravedad en los eslabones puede ser incluido simplemente asignando a la aceleración lineal del eslabón 0, la magnitud del vector de gravedad pero apuntando en la dirección opuesta (${}^0 \dot{v}_0 = G$). Esto es lo equivalente a decir que la base del robot se está acelerando hacia arriba con 1g de aceleración. Esta aceleración ficticia hacia arriba causa exactamente el mismo efecto en los eslabones como lo haría la gravedad. Por consiguiente sin necesidad de un costo extra computacional el efecto de la gravedad es calculado (Craig, 2005).

1.4.2 CONSUMO DE ENERGÍA MANIPULADORES SERIALES

La energía usada en el manipulador se relaciona directamente con el consumo de sus actuadores. Los motores eléctricos son los actuadores más populares usados en los manipuladores. Aunque estos no posean la relación potencia peso que tienen los actuadores hidráulicos o neumáticos, su controlabilidad y su facilidad de interfaz, los hace atractivos para usar en manipuladores de pequeño y mediano tamaños (Craig, 2005).

Los motores de corriente directa (CD) y de imán permanentes, hoy en día son los motores de uso más común en el campo de la robótica (Spong, Hutchinson, & Vidyasagar, 2004). En estos la corriente eléctrica varía con las velocidades y torques sobre ellos. Por consiguiente, en este tipo de motores, aparece el fenómeno de auto-inductancia. La corriente I arrastrada por los motores y la potencial electromotriz de los motores V_e puede calcularse como una función del torque requerido τ_m y la velocidad angular ω de los actuadores, es decir,

$$I = \frac{\tau_m}{K_t} \quad (57)$$

$$V_e = K_e \omega \quad (58)$$

Donde K_t es el factor de sensibilidad del motor o constante de motor expresada en (Nm/A) y K_e la constante de fuerza contra electromotriz expresada en (V*(rad/seg)⁻¹).

La potencia eléctrica total P_T se compone de:

Pérdida de potencia resistiva (efecto Joule):

$$P_J = RI^2 \quad (59)$$

Potencia usada para producir la fuerza electromotriz:

$$P_{EM} = V_e I \quad (60)$$

Por consiguiente, la potencia eléctrica total P_T , puede expresarse como:

$$P_T = P_J + P_{EM} \quad (61)$$

Donde R es la resistencia de la armadura del motor expresada en Ohm.

Finalmente, la energía E consumida por el motor puede ser evaluada mediante la integración de P_T sobre el tiempo total T de la trayectoria:

$$E = \int_0^T P_T dt \quad (62)$$

P_T hace referencia a la potencia eléctrica instantánea consumida en un tiempo t .

Debe tenerse en cuenta que la ecuación (57) de la corriente I , permite considerar la energía utilizada por los actuadores cuando estos no se mueven, pero producen un torque que mantiene el manipulador en una cierta configuración estacionaria (respecto a una dirección particular del actuador), suficiente para resistir la gravedad (Ur-Rehman et al., 2010).

1.4.3 PARÁMETROS DINÁMICOS

El uso de modelos dinámicos para resolver problemas de simulación y control demanda el conocimiento de los valores de los parámetros dinámicos del manipulador.

El mínimo conjunto de parámetros a determinar se conocen como parámetros dinámicos base y su identificación puede reducir significativamente el costo computacional del modelo dinámico. La cantidad de parámetros a conocer depende del modelo dinámico planteado. Para el caso del modelo dinámico del manipulador serial antropomórfico, los parámetros base a determinar son el peso, la ubicación del centro de masa y el tensor de inercia de los eslabones, coeficientes de viscosidad de los motores, constante de torque y constante de fuerza electromotriz de los motores, resistencia de las armaduras de los motores y relación de transmisión de los reductores.

Técnicas de modelado en CAD (Computer Aided Design) pueden ser adoptadas para determinar los valores de los parámetros de inercia de varios componentes (eslabones, actuadores y transmisiones) en base a la geometría y tipo de material empleado. Sin embargo la estimación obtenida con esta técnica puede ser imprecisa por la simplificación realizada típicamente en el modelado geométrico; más aún efectos dinámicos complejos, tales como la fricción en las articulaciones, no pueden ser tomados en cuenta.

Un enfoque heurístico realizado es el de desmontar los diversos componentes del manipulador y realizar una serie de mediciones para evaluar los parámetros de inercia y peso de sus diversos componentes (Siciliano et al., 2009).

Entre estas mediciones la más comunes consisten en pesar el eslabón en una balanza de brazos; encontrar el centro de gravedad mediante el balanceo de cada eslabón en el filo de una cuchilla, una vez ortogonal a cada eje; y los términos de la diagonal del tensor de inercia se miden mediante el método de suspensión con dos cables (Armstrong, Khatib, & Burdick, 1986).

Los coeficientes de viscosidad de los motores se pueden hallar midiendo el promedio de la corriente para varias velocidades de las articulaciones en un rango angular corto, partiendo de una posición vertical inicial. Esto se hace con el fin de eliminar la componente del torque debido a la gravedad. Luego se realiza un gráfico de corriente contra velocidad, y partiendo de un conocimiento previo de la constante de torque del motor de la articulación, la viscosidad se determina a partir de la pendiente de la gráfica. (Corke P. , 1994)

La constante de torque de los motores frecuentemente es medida aplicando un torque conocido a las articulaciones del robot en el modo de control de posición y midiendo la corriente demandada por el sistema del servo para resistir esa carga. Este procedimiento experimental demanda el desmonte del servo del robot, contar con poleas, pesos y software para posicionar o mover los motores del robot.

Otro método para determinar la constante de torque de cada motor de las articulaciones, consiste en desconectar eléctricamente el motor del amplificador de potencia, y el correspondiente eslabón del robot es girado manualmente, mientras el ángulo del eje del motor y el voltaje del motor son grabados. El voltaje de circuito abierto es producido debido a la fuerza contra-electromotriz (fuerza medida en voltios) del imán permanente del motor. Cabe anotar que la constante de torque del motor y la constante de fuerza contra-electromotriz poseen el mismo valor numérico pero difieren en las unidades, siendo Nm/A para la constante de torque y V*seg/rad para la constante contra-electromotriz (Corke P. , 1996).

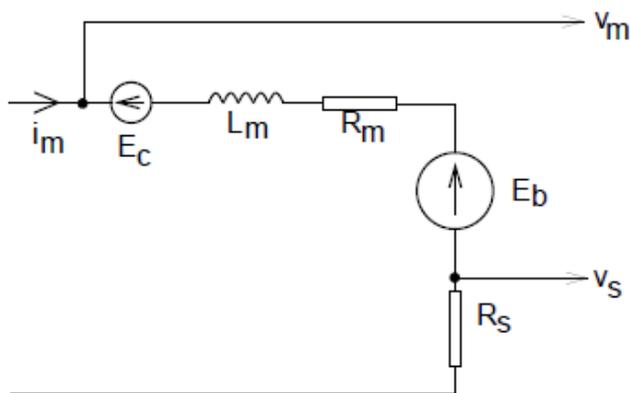


Figura 12 Esquema modelo eléctrico del motor (Corke P. , The Unimation Puma servo system, 1994)

La medida de la resistencia debido a la armadura del motor debe hacerse con este en movimiento, debido a que permite contar con el efecto de la fuerza contra-electromotriz. Partiendo de la ecuación del esquema del modelo eléctrico del motor (Fig. 12):

$$\frac{v_m}{v_s} = \frac{R_m + R_s}{R_s} + \frac{K_m^2 \dot{\theta}}{R_s \tau_m} \quad (63)$$

Donde

v_m : voltaje del motor

v_s : voltaje de circuito derivado

R_s : resistencia del circuito derivado

R_m : resistencia debido a la armadura del motor

K_m : constante de fuerza contra-electromotriz

$\dot{\theta}$: velocidad de la articulación

τ_m : torque del motor

Es posible determinar la resistencia debido a la armadura R_m , conociendo previamente los valores de v_m , v_s , R_s , K_m , $\dot{\theta}$ y τ_m . (Corke P. , The Unimation Puma servo system, 1994).

1.5 MÉTODO DE OPTIMIZACIÓN MEDIANTE ALGORITMO HEURÍSTICO DE KALMAN

En todas las áreas de la ingeniería, física y ciencias sociales, es usual encontrar problema que estén relacionados con la optimización de alguna función objetivo. Generalmente estos problemas son no convexos e inherentemente no lineales y multimodales. Para dar solución a estos problemas se han desarrollado métodos estocásticos, también llamados metaheurísticos. Entre estos métodos metaheurísticos, los más usados en el marco de los problemas de optimización continua son: Los Algoritmos Genéticos (AG), el algoritmo del recocido simulado (Simulated Annealing – SA), y la optimización del enjambre de partículas (Particle Swarm Optimization- PSO). La principal característica de estos enfoques es el uso de mecanismos estocásticos para buscar una solución.

El algoritmo del recocido simulado (Simulated Annealing – SA), es un método de búsqueda aleatoria el cual se destaca porque evitar quedarse atrapado en un mínimo local. Este se obtiene mediante un procedimiento aleatorio, el cual no solo acepta cambios que disminuyen la función

de costo J (asumido como un problema de minimización), sino que adicionalmente permite algunos cambios que la incrementarían. La principal ventaja de este método radica en que permite alcanzar una solución de buena calidad, es decir, el error absoluto para el mínimo global es generalmente menor que el obtenida a través de otros metaheurísticos. Además es versátil y fácil de implementar. El principal inconveniente de este método radica en la elección de los diversos parámetros que intervienen en este algoritmo, siendo los resultados obtenidos muy sensibles a la configuración de estos parámetros. Por consiguiente el problema de la selección de unos buenos parámetros (para una función de costo dada) es un tema crucial, el cual no ha sido totalmente resuelto. Otra debilidad de este método ligada al problema de la configuración de los parámetros, es el tiempo excesivo de cómputo en la mayoría de las aplicaciones.

El algoritmo genético (AG), es una técnica de búsqueda estocástica basado en la población. Este método utiliza poblaciones de puntos que contienen varias soluciones posibles, cada uno de las cuales se evalúa y una nueva población es creada a partir de la mejor de estas mediante operadores aleatorios, como la selección, cruce y mutación, inspirados en la reproducción natural y la evolución de los seres vivos. El proceso continúa a través de un número de generaciones (es decir, iteraciones) con el objetivo de lograr que la población evolucione hacia una solución aceptable. La principal ventaja del AG (y de sus muchas versiones) es su robustez, así como su intuitividad, la facilidad de implementación y la capacidad de enfrentar con éxito una amplia gama de problemas difíciles. Un inconveniente principal con los AG es que algunos de los individuos bien adaptados (en comparación con otros miembros de la población, pero lejano desde el punto óptimo), dominan la población, haciendo que converjan en un mínimo local. En estas condiciones, la probabilidad de encontrar buenas soluciones es muy pequeña porque el cruce entre individuos similares, produce cambios pequeños. Sólo la mutación sigue buscando los mejores individuos, pero esto no suele ser suficiente para una rápida convergencia hacia la mejor solución. Esto último requiere por lo tanto un tiempo excesivo de cómputo.

Los AG y la optimización del enjambre de partículas (Particle Swarm Optimization- PSO), son similares, debido a que estos dos enfoques están basados en métodos de búsqueda aleatoria de población pero con diferentes estrategias de evolución. La optimización del enjambre de partículas se inspira en el comportamiento colectivo de los seres vivos, incluyendo la noción de inteligencia colectiva de una población de individuos. Este es un algoritmo de búsqueda basado en la población, donde cada individuo se denomina partícula y representa un candidato a solución. Cada partícula evoluciona a través del espacio de búsqueda, con el fin de hallar la solución óptima del problema de optimización. La principal ventaja del PSO es su facilidad de aplicación, así como su capacidad para encontrar buenas soluciones mucho más rápido que otras metaheurísticos (menor número de funciones evaluadas). Sin embargo, este no puede mejorar la calidad de las soluciones a medida que aumenta el número de iteraciones. Al igual que el AG, un inconveniente importante con el PSO, es que el enjambre puede converger prematuramente, no siendo esta convergencia necesariamente un mínimo local. Otro inconveniente, es la gran sensibilidad del PSO a los ajustes de parámetros: un cambio pequeño en los parámetros puede dar lugar a un efecto proporcionalmente grande (Toscano, 2010).

Siguiendo con la línea de estos enfoques de optimización, el algoritmo heurístico de Kalman (Heuristic Kalman Algorithm - HKA) es propuesto por (Toscano & Lyonnet, 2010). En este método la búsqueda heurística es totalmente diferente a la de los métodos mencionados anteriormente. De hecho, el HKA considera explícitamente el problema de la optimización, como un proceso de medición destinado a dar un estimado del óptimo. Este utiliza una función de densidad de probabilidad Gaussiana (FDPG), como proceso de medición, y un estimador de Kalman que permite mejorar la calidad de la estimación obtenida a través del proceso de medición. La FDPG evoluciona en el espacio de búsqueda, tratando de encontrar la solución óptima del problema de optimización. Una FDPG se caracteriza por su vector de medias m , y su matriz de varianzas Σ . Para la búsqueda de la solución óptima, los parámetros de la FDPG son actualizados, teniendo en cuenta los puntos de muestra obtenidos a través de un proceso de medición, esto se hace usando un estimador de Kalman. De hecho el estimador de Kalman puede ser visto como un mecanismo capaz de actualizar el conocimiento acerca de las cantidades desconocidas de interés, teniendo en cuenta la nueva información adquirida. El "movimiento" de la FDPG se ajusta en función de su valor medio actual y la nueva información obtenida a través del proceso de medición. La repetición de este procedimiento conduce a la FDPG hacia un dominio del espacio de búsqueda que contiene las soluciones de alta calidad. HKA comparte con algunos de los algoritmos estocásticos las mismas características interesantes, tales como: facilidad de implementación, la memoria baja y los requisitos de velocidad de procesador, procedimientos de búsqueda basados únicamente en los valores de la función objetivo, sin necesidad de suposiciones fuertes tales como linealidad, diferenciabilidad, convexidad, etc., para resolver el problema de optimización. De hecho, podría ser utilizado incluso cuando la función objetivo no puede ser expresada en forma analítica, en este caso, la función objetivo se evalúa a través de simulaciones. Sin embargo, el principal inconveniente radica en que el HKA puede converger prematuramente en una solución local, en particular cuando el coeficiente de ralentización es demasiado alto.

1.5.1 PRINCIPIO DEL ALGORITMO HKA

El principio del algoritmo es ilustrado en la Fig. 13. El procedimiento propuesto es iterativo, y se denota por, la k -ésima iteración del algoritmo. Este cuenta con un generador aleatorio de funciones de probabilidad (PDF) $g(k)$, la cual produce, en cada iteración una colección de N vectores que son distribuidos a lo largo de un vector de medias dado $m(k)$ con una matriz de varianza-covarianza $\Sigma(k)$ dada. Esta recopilación puede escribirse de la siguiente forma:

$$q(k) = \{q^1(k), q^2(k), \dots, q^N(k)\} \quad (64)$$

Donde $q^i(k)$ es el vector i -ésimo generado en la iteración número k : $q^i(k) = [q_1^i(k), \dots, q_{n_q}^i(k)]^T$, y $q_l^i(k)$ es el l -ésimo componente de $q^i(k) = (l = 1, \dots, n_q)$.

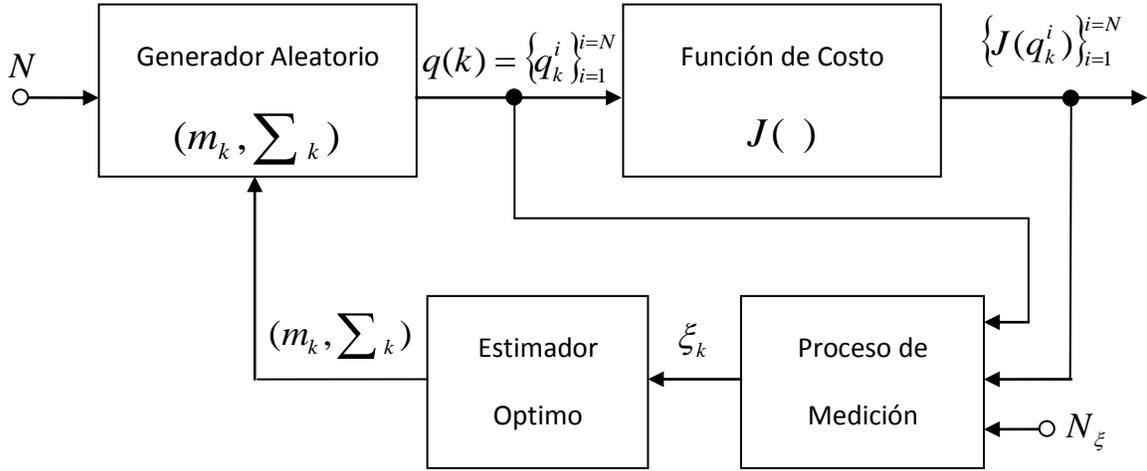


Figura 13 Principio del algoritmo HKA

Este generador aleatorio es aplicado a la función de costo J . Sin pérdida de generalidad, se parte de la suposición que los vectores están ordenados por su función de costo creciente, es decir:

$$J(q^1(k)) < J(q^2(k)) < \dots < J(q^N(k)) \quad (65)$$

El principio del algoritmo consiste en modificar el vector de medias y la matriz de varianzas del generador aleatorio hasta que una solución de alta calidad sea obtenida. Más precisamente, N_ξ es el número de mejores muestras consideradas, tales que $J(q^{N_\xi}(k)) < J(q^i(k))$ para todo $i > N_\xi$. Cabe anotar que las mejores muestras son aquellas de la secuencia (ecuación 65) que poseen las menores funciones de costo. El objetivo es entonces generar, de las mejores muestras, una nueva distribución aleatoria que se aproxime al mínimo de la función de costo J . El problema es como modificar los parámetros del generador aleatorio para alcanzar una estimación confiable del óptimo.

Para resolver este problema, se introduce un procedimiento de medición seguido por un estimador óptimo de los parámetros del generador aleatorio. El proceso de medición consiste en calcular el promedio de los candidatos que son los más representativos del óptimo. Para la iteración k , la medida, denotada $\xi(k)$, es definida como sigue:

$$\xi(k) = \frac{1}{N_\xi} \sum_{i=1}^{N_\xi} q^i(k) \quad (66)$$

Donde N_ξ es el número de mejores muestras consideradas. Esta medida da un conocimiento afectado por el óptimo, es decir:

$$\xi(k) = q_{opt} + v(k) \quad (67)$$

Donde $v(k)$ es una perturbación desconocida, la cual se centra en q_{opt} , y actúa sobre la medición del proceso. Cabe anotar que $v(k)$ es un tipo de medida de la ignorancia sobre q_{opt} . Esta incertidumbre no se puede medir, pero sólo se estima teniendo en cuenta todo el conocimiento disponible. Para este caso, la incertidumbre de la medida está estrechamente relacionada con la dispersión de las mejores muestras $q^i(k) = (i = 1, \dots, N_\xi)$.

La ignorancia sobre el óptimo, puede ser tenida en cuenta usando el vector de varianza asociado a las mejores muestras:

$$V(k) = \frac{1}{N_\xi} \left[\sum_{i=1}^{N_\xi} (q_1^i(k) - \xi_1(k)), \dots, \sum_{i=1}^{N_\xi} (q_{n_q}^i(k) - \xi_{n_q}(k)) \right]^T \quad (68)$$

En estas condiciones, el estimador de Kalman se puede utilizar para hacer una estimación, los llamados "a posteriori", del óptimo, es decir, teniendo en cuenta la medida, así como la confianza puesta en ella. Esta confianza puede ser cuantificada por el vector de varianza (ecuación 68).

Para diseñar un estimador óptimo que combine una estimación previa de q_{opt} y la medida $\xi(k)$, razón por la cual la estimación posterior resultante será mejor en el sentido de una disminución de la función de costo (problema de minimización). En base a las ecuaciones de Kalman, la regla de actualización del generador Gaussiano es de la siguiente manera:

$$\begin{cases} m(k+1) = m(k) + L(k)(\xi(k) - m(k)) \\ \Sigma(k+1) = (I - a(k)L(k))\Sigma(k) \end{cases} \quad (69)$$

Con:

$$L(k) = \Sigma(k) \left(\Sigma(k) + D(k) \right)^{-1}, \text{ y} \quad (70)$$

$$a(k) = \frac{\alpha \min \left(1, \left(\frac{1}{n_q} \sum_{i=1}^{n_q} \sqrt{v_i(k)} \right)^2 \right)}{\min \left(1, \left(\frac{1}{n_q} \sum_{i=1}^{n_q} \sqrt{v_i(k)} \right)^2 \right) + \max_{1 \leq i \leq n_q} (v_i(k))} \quad (71)$$

Donde $D(k)$ es una matriz diagonal que tiene en su diagonal el vector de varianza $V(k)$, $v_i(k)$ representa la i -ésima componente del vector de varianza $V(k)$ y un coeficiente de ralentización $\alpha \in (0,1]$ dado por el usuario. El coeficiente $a(k)$ se usa para controlar la disminución en el tiempo de la matriz de varianza $\Sigma(k)$. Esta disminución asegura una transición progresiva de la búsqueda global a la búsqueda local (Toscano & Lyonnet, 2010).

1.5.2 DESARROLLO DEL ALGORITMO HKA

A continuación se describen los pasos del algoritmo para realizar la minimización de la función objetivo $J(q)$.

Paso Inicialización. Escoger N , N_ξ y α . Estableciendo $k = 0$, $m(k) = m_0$, $\sum(k) = \sum_0$.

Paso Generador Gaussiano. Generar una secuencia de N vectores $q^1(k), q^2(k), \dots, q^N(k)$, de acuerdo a la distribución Gaussiana parametrizada por $m(k)$ y $\sum(k)$.

Paso Proceso de medición. Calcular $\xi(k)$ y $V(k)$.

Paso Actualizar la regla del generador Gaussiano.

Paso Regla de paro. Si la regla de paro no se satisface ir al paso del Generador Gaussiano, de otra manera se detiene en el paso del Generador Gaussiano.

La inicialización y ajuste de parámetros del generador Gaussiano, debe hacerse cubriendo el espacio completo de búsqueda. Para tal fin, la siguiente regla puede ser usada:

$$m_0 = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{n_q} \end{bmatrix}, \sum_0 = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_{n_q} \end{bmatrix}, \text{ con } \begin{cases} \mu_i = \frac{\bar{q}_i + \underline{q}_i}{2} \\ \sigma_i = \frac{\bar{q}_i - \underline{q}_i}{6} \end{cases}, i = 1, \dots, n_q \quad (72)$$

Donde \bar{q}_i es el i -ésimo límite superior y \underline{q}_i es el i -ésimo límite inferior de la hipercaja del dominio de búsqueda. Con esta regla, 99% de las muestras se generan en los intervalos: $\mu_i \pm 3\sigma_i$, $i = 1, \dots, n_q$.

Para el algoritmo se deben establecer los tres parámetros siguientes: el número de puntos N , el número de mejores candidatos N_ξ y el coeficiente de ralentización α . Para facilitar esta tarea, en la tabla 1 resume el ajuste de los parámetros estándar de HKA.

Número de puntos de la muestra (N)	$20 \leq N \leq 150$
Número de mejores candidatos (N_ξ)	$2 \leq N_\xi \leq N$
Coficiente de ralentización (α)	$0.1 \leq \alpha \leq 0.9$

Tabla 1 Ajuste de los parámetros estándar de HKA

La colección de los N vectores puede ser escogida con el fin de generar al menos N_ξ muestras dentro de una bola con radio conocido. Este radio se conoce como el radio medio Gaussiano (ρ_k^m), y este es tal que 1/2 de la masa de probabilidad del Gaussiano cae dentro de una bola de radio ρ_k^m . Este satisface la condición $\rho_k^m \leq \sqrt{2tr(\sum(k))}$. Consecuentemente la probabilidad p_B para dibujar una muestra dentro de una bola de radio $\sqrt{2tr(\sum(k))}$ centrada en $m(k)$ es $p_B \geq 1/2$. Además. Para N muestras la probabilidad $p_B^{N_\xi}$ para dibujar al menos N_ξ muestras en $B(m(k), \sqrt{2tr(\sum(k))})$, está dada por (Toscano & Lyonnet, 2009):

$$p_B^{N_\xi} = \sum_{i=N_\xi}^N \frac{N!}{i!(N-i)!} p_B^i (1-p_B)^{N-i} \quad (73)$$

Tomando el límite inferior de p_B , se tiene que:

$$p_B^{N_\xi} \geq \frac{1}{2^N} \sum_{i=N_\xi}^N \frac{N!}{i!(N-i)!} \quad (74)$$

Esta relación puede ser usada para determinar el número total de muestras N requeridas para obtener, con una probabilidad dada, al menos N_ξ muestras dentro de la bola $B(m(k), \sqrt{2tr(\sum(k))})$.

La regla de paro del algoritmo se determina fijando un número máximo de iteraciones o estableciendo un indicador de precisión.

1.6 PRODUCTIVIDAD

La productividad es la relación aritmética entre la cantidad producida (salida) y la cantidad de recursos usados (entrada) para realizar la producción. La productividad puede ser expresada como:

$$productividad = \frac{Salidas}{entradas} \quad (75)$$

La productividad hace referencia a la eficiencia de los sistemas de producción. Es un indicador de lo bien que los factores de producción (tierra, capital, mano de obra y energía) han sido usados.

Los factores que influyen la productividad pueden ser clasificados generalmente en dos categorías: factores controlables o internos y factores no controlables o externos.

Los factores controlables son: el factor de producto, es decir como el producto satisface los requerimientos del usuario, la disponibilidad de la planta o equipo usado, la tecnología disponible, los materiales y energía, el factor humano, los métodos de trabajo, y el estilo gerencial.

Los factores no controlables, hacen referencia a los ajustes estructurales, es decir ajustes, sociales y económicos, recursos naturales, y de gobierno o infraestructura.

La medida de la productividad puede clasificarse como medida de productividad total y medida de productividad parcial:

La medida de productividad total está basada en todas las posibles entradas que afectan la producción dentro de una organización.

Dependiendo de la entrada individual parcial observada, la medida de productividad parcial puede ser expresada como:

$$\textit{productividad por trabajo} = \frac{\textit{total de salidas}}{\textit{trabajo de entrada}} \quad (76)$$

$$\textit{productividad por capital} = \frac{\textit{total de salidas}}{\textit{capital de entrada}} \quad (77)$$

$$\textit{productividad por material} = \frac{\textit{total de salidas}}{\textit{material de entrada}} \quad (78)$$

$$\textit{productividad por energía} = \frac{\textit{total de salidas}}{\textit{energía de entrada}} \quad (79)$$

Entre las variables tomadas generalmente como salidas de productividad destacan, las unidades fabricadas o procesadas, número de servicios prestados, cantidad de labor realizada o producción atendida, entre otras (Panneerselvam, 2012).

2 MODELADO MATEMÁTICO DE ECUACIONES PARA PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

De los manipuladores disponibles a nivel comercial, los manipuladores antropomórficos de seis grados de libertad con muñeca esférica son los más populares en la industria gracias a su simplicidad y robustez mecánica (Pires, 2007). Cabe anotar que la mayoría de los grandes fabricantes de manipuladores soldadores de arco de seis grados de libertad basan muchos de sus diseños cinemáticos en manipuladores antropomórficos de seis grados de libertad (ABB Robotics, FANUC Robotics, KUKA Robot Group, Motoman Robotics, Panasonic Robot & Welding Systems, Stäubli Robotics).

En este capítulo se expondrán las ecuaciones pertinentes para implementar el método de planeación de trayectorias en manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica aplicados a los procesos de soldadura por arco, basado en el algoritmo heurístico de Kalman, enfocado en la minimización del criterio de consumo eléctrico y maximización de la manipulabilidad.

2.1 CINEMÁTICA DIRECTA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

Para proceder con el cálculo de la cinemática directa de un manipulador se debe inicialmente conocer su modelo cinemático y las dimensiones de los eslabones. Partiendo del esquema del manipulador antropomórfico de seis grados de libertad con muñeca esférica representado en la Fig. 14 se procede a plantear su modelo cinemático basado en la convención DH. En la Tabla 2 se observan los valores hallados para este modelo.

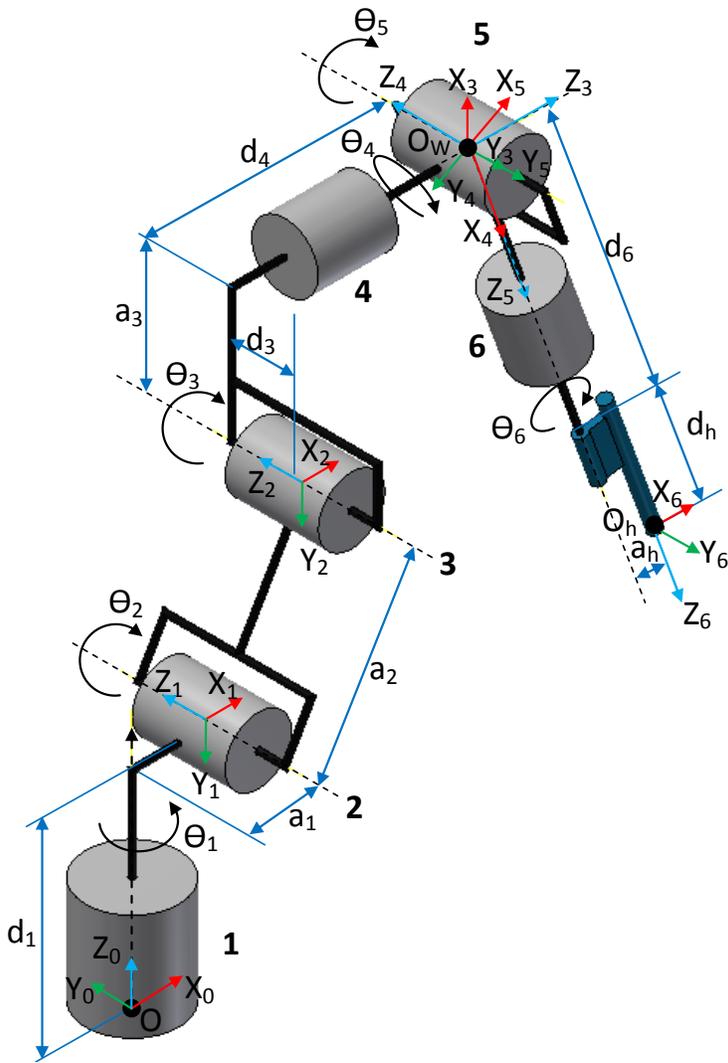


Figura 14 Esquema convención DH robot antropomórfico de seis grados de libertad con muñeca esférica

Articulaciones (θ_i)	d_i (m)	a_i (m)	α_i ($^\circ$)
θ_1	d_1	a_1	-90
θ_2	0	a_2	0
θ_3	d_3	a_3	-90
θ_4	d_4	0	90
θ_5	0	0	-90
θ_6	d_6+d_h	a_h	0

Tabla 2 Modelo cinemático basado en la convención DH para manipulador antropomórfico de seis grados de libertad con muñeca esférica

A partir del modelo cinemático planteado, se hallan las matrices de transformación homogénea para cada articulación:

$${}^0A_1 = \begin{bmatrix} \text{Cos}(\theta_1) & 0 & -\text{Sen}(\theta_1) & a_1\text{Cos}(\theta_1) \\ \text{Sen}(\theta_1) & 0 & \text{Cos}(\theta_1) & a_1\text{Sen}(\theta_1) \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (80)$$

$${}^1A_2 = \begin{bmatrix} \text{Cos}(\theta_2) & -\text{Sen}(\theta_2) & 0 & a_2\text{Cos}(\theta_2) \\ \text{Sen}(\theta_2) & \text{Cos}(\theta_2) & 0 & a_2\text{Sen}(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (81)$$

$${}^2A_3 = \begin{bmatrix} \text{Cos}(\theta_3) & 0 & -\text{Sen}(\theta_3) & a_3\text{Cos}(\theta_3) \\ \text{Sen}(\theta_3) & 0 & \text{Cos}(\theta_3) & a_3\text{Sen}(\theta_3) \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (82)$$

$${}^3A_4 = \begin{bmatrix} \text{Cos}(\theta_4) & 0 & \text{Sen}(\theta_4) & 0 \\ \text{Sen}(\theta_4) & 0 & -\text{Cos}(\theta_4) & 0 \\ 0 & 1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (83)$$

$${}^4A_5 = \begin{bmatrix} \text{Cos}(\theta_5) & 0 & -\text{Sen}(\theta_5) & 0 \\ \text{Sen}(\theta_5) & 0 & \text{Cos}(\theta_5) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (84)$$

$${}^5A_6 = \begin{bmatrix} \text{Cos}(\theta_6) & -\text{Sen}(\theta_6) & 0 & a_h\text{Cos}(\theta_6) \\ \text{Sen}(\theta_6) & \text{Cos}(\theta_6) & 0 & a_h\text{Sen}(\theta_6) \\ 0 & 0 & 1 & d_6 + d_h \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (85)$$

De las matrices anteriores se puede concluir que para manipuladores seriales antropomórficos con seis grados de libertad, y muñeca esférica la matriz que considera todas las articulaciones del robot y permite identificar la posición y orientación del efector final puede enunciarse como:

$$T = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 {}^5A_6 \quad (86)$$

$$T = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (87)$$

Dónde:

n: Vector normal del efector final o herramienta

s: Vector de deslizamiento del efector final o herramienta.

a: Vector de aproximación

p: Vector de posición del efector final o herramienta.

Y son iguales a:

$$\begin{aligned} n_x = & \text{Sen}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Sen}(\Theta_6) + \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_4)) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_1) * (\text{Sen}(\Theta_4) * \text{Sen}(\Theta_6) - \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) - \text{Sen}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_1) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_5) \end{aligned} \quad (88)$$

$$\begin{aligned} s_x = & \text{Sen}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Cos}(\Theta_6) - \text{Cos}(\Theta_5) * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_6)) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_1) * (\text{Cos}(\Theta_6) * \text{Sen}(\Theta_4) + \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Sen}(\Theta_6)) + \text{Sen}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_1) * \text{Sen}(\Theta_5) * \text{Sen}(\Theta_6) \end{aligned} \quad (89)$$

$$\begin{aligned} a_x = & - \text{Sen}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_1) * \text{Cos}(\Theta_5) - \text{Sen}(\Theta_1) * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_5) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_1) * \text{Cos}(\Theta_4) * \text{Sen}(\Theta_5) \end{aligned} \quad (90)$$

$$\begin{aligned} p_x = & a_1 * \text{Cos}(\Theta_1) - d_3 * \text{Sen}(\Theta_1) + \text{Sen}(\Theta_1) * (a_h * \text{Cos}(\Theta_4) * \text{Sen}(\Theta_6) - \text{Sen}(\Theta_4) * \text{Sen}(\Theta_5) * (d_6 \\ & + d_h) + a_h * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_4)) - \text{Cos}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Sen}(\Theta_5) * (d_6 \\ & + d_h) + a_h * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_6) - a_h * \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) + a_2 * \text{Cos}(\Theta_1) * \text{Cos}(\Theta_2) - \\ & \text{Sen}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_1) * (d_4 + \text{Cos}(\Theta_5) * (d_6 + d_h) + a_h * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_5)) - \\ & a_3 * \text{Cos}(\Theta_1) * \text{Sen}(\Theta_2) * \text{Sen}(\Theta_3) + a_3 * \text{Cos}(\Theta_1) * \text{Cos}(\Theta_2) * \text{Cos}(\Theta_3) \end{aligned} \quad (91)$$

$$\begin{aligned} n_y = & - \text{Cos}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Sen}(\Theta_6) + \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_4)) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Sen}(\Theta_1) * (\text{Sen}(\Theta_4) * \text{Sen}(\Theta_6) - \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) - \text{Sen}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_1) * \text{Sen}(\Theta_5) \end{aligned} \quad (92)$$

$$\begin{aligned} s_y = & \text{Sen}(\Theta_2 + \Theta_3) * \text{Sen}(\Theta_1) * \text{Sen}(\Theta_5) * \text{Sen}(\Theta_6) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Sen}(\Theta_1) * (\text{Cos}(\Theta_6) * \text{Sen}(\Theta_4) + \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Sen}(\Theta_6)) - \\ & \text{Cos}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Cos}(\Theta_6) - \text{Cos}(\Theta_5) * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_6)) \end{aligned} \quad (93)$$

$$\begin{aligned} a_y = & \text{Cos}(\Theta_1) * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_5) - \text{Sen}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_5) * \text{Sen}(\Theta_1) - \text{Cos}(\Theta_2 + \\ & \Theta_3) * \text{Cos}(\Theta_4) * \text{Sen}(\Theta_1) * \text{Sen}(\Theta_5) \end{aligned} \quad (94)$$

$$\begin{aligned} p_y = & d_3 * \text{Cos}(\Theta_1) + a_1 * \text{Sen}(\Theta_1) - \text{Cos}(\Theta_1) * (a_h * \text{Cos}(\Theta_4) * \text{Sen}(\Theta_6) - \text{Sen}(\Theta_4) * \text{Sen}(\Theta_5) * (d_6 \\ & + d_h) + a_h * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_4)) - \text{Cos}(\Theta_2 + \Theta_3) * \text{Sen}(\Theta_1) * (\text{Cos}(\Theta_4) * \text{Sen}(\Theta_5) * (d_6 \\ & + d_h) + a_h * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_6) - a_h * \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) + a_2 * \text{Cos}(\Theta_2) * \text{Sen}(\Theta_1) - \\ & \text{Sen}(\Theta_2 + \Theta_3) * \text{Sen}(\Theta_1) * (d_4 + \text{Cos}(\Theta_5) * (d_6 + d_h) + a_h * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_5)) - \\ & a_3 * \text{Sen}(\Theta_1) * \text{Sen}(\Theta_2) * \text{Sen}(\Theta_3) + a_3 * \text{Cos}(\Theta_2) * \text{Cos}(\Theta_3) * \text{Sen}(\Theta_1) \end{aligned} \quad (95)$$

$$\begin{aligned} n_z = & \text{Sen}(\Theta_2 + \Theta_3) * (\text{Sen}(\Theta_4) * \text{Sen}(\Theta_6) - \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) - \text{Cos}(\Theta_2 + \end{aligned} \quad (96)$$

$$\Theta_3) * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_5)$$

$$\mathbf{s}_z = \text{Sen}(\Theta_2 + \Theta_3) * (\text{Cos}(\Theta_6) * \text{Sen}(\Theta_4) + \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Sen}(\Theta_6)) + \text{Cos}(\Theta_2 + \Theta_3) * \text{Sen}(\Theta_5) * \text{Sen}(\Theta_6) \quad (97)$$

$$\mathbf{a}_z = \text{Sen}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_4) * \text{Sen}(\Theta_5) - \text{Cos}(\Theta_2 + \Theta_3) * \text{Cos}(\Theta_5) \quad (98)$$

$$\mathbf{p}_z = d_1 - \text{Cos}(\Theta_2 + \Theta_3) * (d_4 + \text{Cos}(\Theta_5) * (d_6 + d_h) + a_h * \text{Cos}(\Theta_6) * \text{Sen}(\Theta_5)) - a_3 * \text{Sen}(\Theta_2 + \Theta_3) + \text{Sen}(\Theta_2 + \Theta_3) * (\text{Cos}(\Theta_4) * \text{Sen}(\Theta_5) * (d_6 + d_h) + a_h * \text{Sen}(\Theta_4) * \text{Sen}(\Theta_6)) - a_h * \text{Cos}(\Theta_4) * \text{Cos}(\Theta_5) * \text{Cos}(\Theta_6)) - a_2 * \text{Sen}(\Theta_2) \quad (99)$$

2.2 CINEMÁTICA INVERSA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

Por medio del desarrollo de la cinemática inversa se busca hallar los valores de las seis variables articulares del manipulador partiendo de una ubicación y orientación conocida del efector final o herramienta del manipulador. Para los manipuladores antropomórficos de seis grados de libertad con muñeca esférica es posible desacoplar el problema de la cinemática inversa en dos problemas independientes: uno enfocado en hallar la posición de la intersección de los ejes de la muñeca, conocido como centro de la muñeca (O_w), y otro en el cual se halla la orientación de la muñeca.

El punto importante de realizar este desacople en un manipulador con muñeca esférica, es que el movimiento de los tres últimos eslabones alrededor de sus ejes no cambiará la posición del centro de muñeca O_w , debido a que la posición de este es solo función de las tres primeras variables de articulación.

Tomando como base la acotación anterior, y conociendo el punto de la posición del efector final o herramienta (O_h) respecto a la base del robot (O), es posible hallar la distancia de centro de muñeca respecto a las coordenadas de la base ($\overline{OO_w}$), desplazándose un ΔH compuesto por una distancia $d_6 + d_h$ respecto a su eje Z, y una distancia a_h respecto a su eje X, desde el punto final del efector final o herramienta (O_h) Fig. 15. De lo anterior se tiene que:

$$\overline{OO_w} = \overline{OO_h} - \Delta H * T = \begin{cases} \overline{OO_{wx}} = p_x + a_6 * n_x + (d_6 + d_h) * a_x \\ \overline{OO_{wy}} = p_y + a_6 * n_y + (d_6 + d_h) * a_y \\ \overline{OO_{wz}} = p_z + a_6 * n_z + (d_6 + d_h) * a_z \end{cases} \quad (100)$$

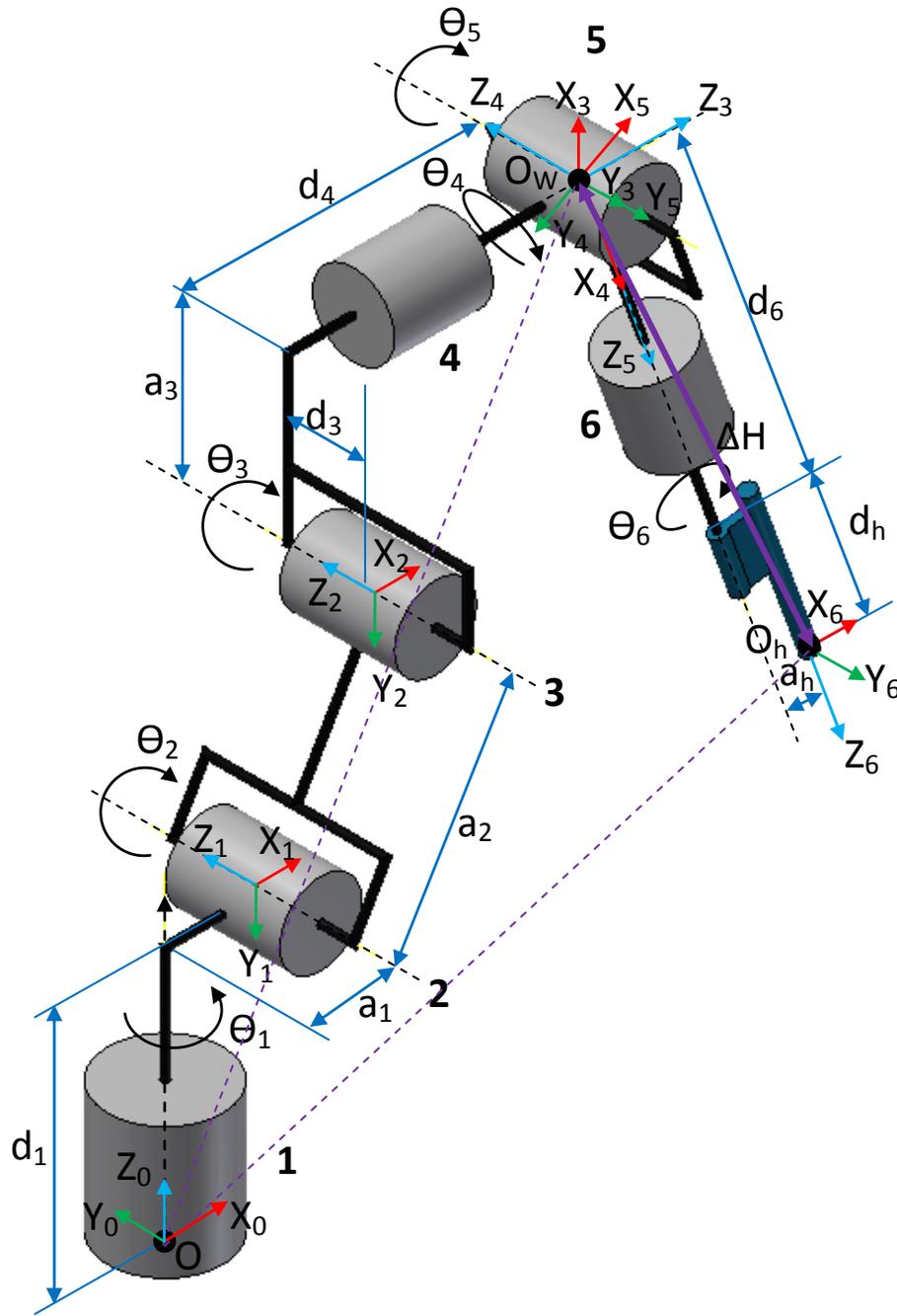


Figura 15 Esquema para cinemática inversa de robot antropomórfico de seis grados de libertad con muñeca esférica

Partiendo de la ecuación 100, y realizando un análisis geométrico con el apoyo de una representación gráfica (Fig. 16), se procede a hallar los valores de las tres primeras variables articulares.

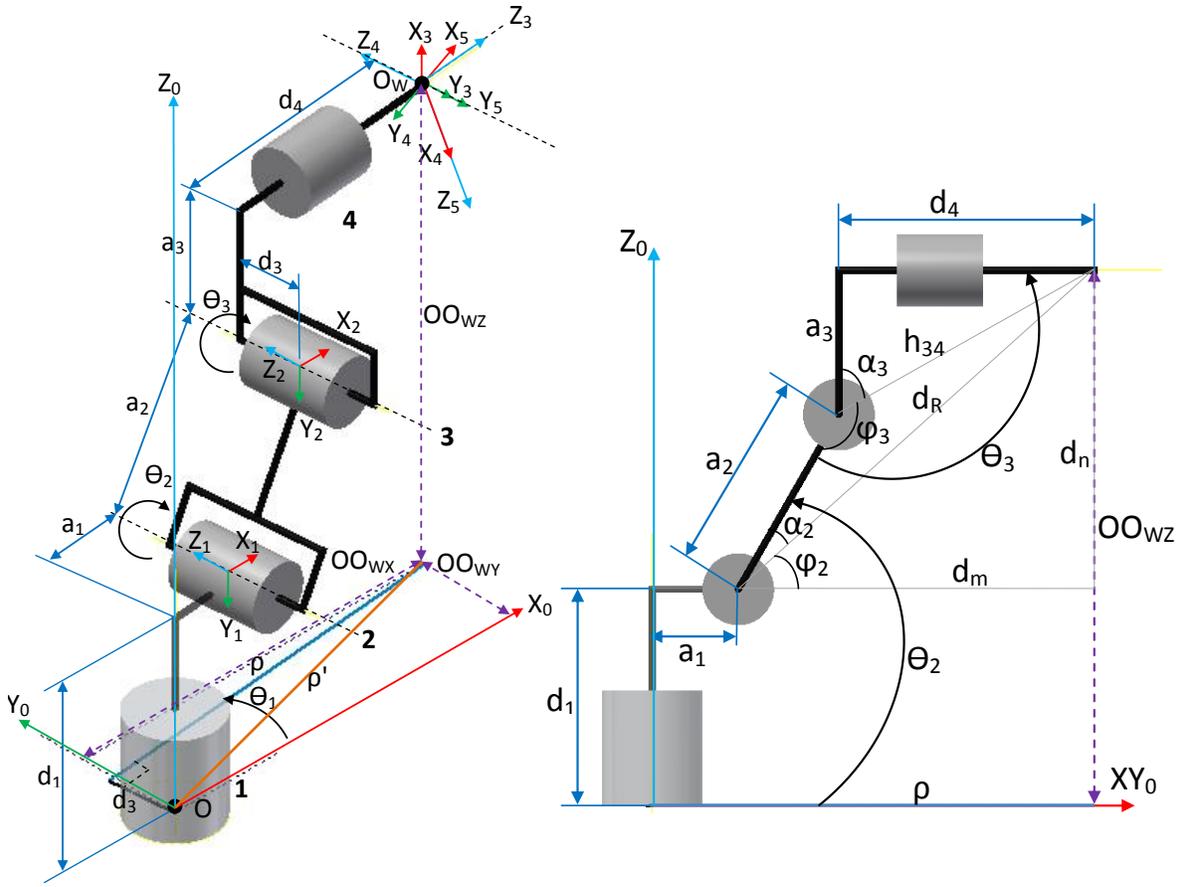


Figura 16 Esquema tres primeros eslabones robot antropomórfico de seis grados de libertad con muñeca esférica

De la Fig. 16 se deduce que:

$$\rho' = \sqrt{(OO_{wx})^2 + (OO_{wy})^2} \quad (101)$$

$$\rho = \sqrt{(\rho')^2 - (d_3)^2} \quad (102)$$

$$\theta_1 = \arctan 2(OO_{wx}, OO_{wy}) - \arcsin(d_3/\rho') \quad (103)$$

$$d_m = \rho - a_1 \quad (104)$$

$$d_n = OO_{wz} - d_1 \quad (105)$$

$$d_R = \sqrt{d_m^2 + d_n^2} \quad (106)$$

$$\varphi_2 = \arctan 2(d_m, d_n) \quad (107)$$

$$\alpha_2 = \arccos \left(\frac{d_R^2 + a_2^2 - h_{34}^2}{2 * a_2 * d_R} \right) \quad (108)$$

$$\theta_2 = \alpha_2 + \varphi_2 \quad (109)$$

$$\alpha_3 = \arctan \left(\frac{d_4}{a_3} \right) \quad (110)$$

$$\varphi_3 = \arccos \left(\frac{a_2^2 + h_{34}^2 - d_R^2}{2a_2 * h_{34}} \right) \quad (111)$$

$$\theta_3 = 90 - (\alpha_3 + \varphi_3) \quad (112)$$

Siendo θ_1 , θ_2 , y θ_3 las tres primeras variables articulares del manipulador, halladas en las ecuaciones 103, 109 y 112 respectivamente. Estas tres variables articulares se usan para hallar la matriz de orientación 0R_3 la cual en conjunto con la matriz de orientación del efector final R , serán la base para calcular las tres variables articulares restantes.

Partiendo de:

$${}^0R_3 = {}^0R_1 {}^1R_2 {}^2R_3 \quad (113)$$

$$R = {}^0R_3 {}^3R_6 \quad (114)$$

$${}^3R_6 = ({}^0R_3)^{-1} R = ({}^0R_3)^T R = [r_{ij}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (115)$$

Y Conociendo además que 3R_6 simbólicamente es igual a:

$${}^3R_6 = {}^3R_4 {}^4R_5 {}^5R_6 \quad (116)$$

$${}^3R_6 = \begin{bmatrix} \cos\theta_4 \cos\theta_5 \cos\theta_6 - \sin\theta_4 \sin\theta_6 & -\cos\theta_4 \cos\theta_5 \sin\theta_6 - \sin\theta_4 \cos\theta_6 & -\cos\theta_4 \sin\theta_5 \\ \sin\theta_4 \cos\theta_5 \cos\theta_6 + \cos\theta_4 \sin\theta_6 & -\sin\theta_4 \cos\theta_5 \sin\theta_6 + \cos\theta_4 \cos\theta_6 & -\sin\theta_4 \sin\theta_5 \\ \sin\theta_5 \cos\theta_6 & -\sin\theta_5 \cos\theta_6 & \cos\theta_5 \end{bmatrix} \quad (117)$$

Igualando ecuaciones 115 y 117 se pueden hallar las variables articulares de los últimos eslabones del manipulador, o de la sección de la muñeca:

$$\theta_4 = \arccos \left(-\frac{r_{13}}{\sqrt{1-r_{33}^2}} \right), \text{ o } \theta_4 = \arccos \left(-\frac{r_{13}}{\sqrt{1-r_{33}^2}} \right) \pm \pi \quad (118)$$

$$\theta_5 = \arccos (r_{33}) - \pi, \text{ o } \theta_5 = -(\arccos (r_{33}) - \pi) \quad (119)$$

$$\theta_6 = \arccos \left(\frac{r_{31}}{\sqrt{1-r_{33}^2}} \right), \text{ o } \theta_6 = \arccos \left(\frac{r_{31}}{\sqrt{1-r_{33}^2}} \right) \pm \pi \quad (120)$$

Cabe resaltar que debido a los seis grados de libertad que poseen estos manipuladores en algunos casos para que el efector final se ubique en un determinado punto en el espacio de trabajo puede disponer de otras tres posibles configuraciones adicionales de sus articulaciones. Estas configuraciones se presentan cuando:

A la variable articular θ_1 se le suma un ángulo de π rad y $\theta_2 = -[180 - (\alpha_2 + \varphi_2)]$.

La variable articular $\theta_2 = \alpha_2 - \varphi_2$

Y una combinación de los casos anteriores, es decir a θ_1 se le suma un ángulo de π rad y $\theta_2 = -[180 - (\alpha_2 - \varphi_2)]$.

2.3 CINEMÁTICA DIFERENCIAL MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

2.3.1 CÁLCULO VELOCIDADES ARTICULARES

Debido a que los manipuladores seriales antropomórficos de seis grados de libertad con muñeca esférica permiten el desacople de su cadena cinemática, es más conveniente tratar el problema de la cinemática diferencial respecto al centro de su muñeca (O_w).

Para ello se define un sistema de coordenadas con origen O_{i-1} , siendo $i = 1, 2, \dots, 6$, fijo a cada eslabón, y se nombran las primeras tres articulaciones que conforman la sección del brazo como n_{A1} , n_{A2} , n_{A3} , y las tres correspondientes a la muñeca como n_{W4} , n_{W5} , n_{W6} . Para este caso el efector final coincide con el último eslabón, por lo que el origen O_h es el punto central del efector final. Adicionalmente se aprecia que los ejes de las articulaciones de la sección de la muñeca se intersecan en el origen O_w . Además p_i denota los vectores de O_{i-1} (con $i = 1$ a 3) hasta O_h , y p_w representa el vector de O_w a O_h . Lo anterior es ilustrado en la Fig. 17.

Tomando la velocidad de la muñeca v_W como punto de partida para el desarrollo de la cinemática diferencial, se tiene que:

$$v_W = \begin{bmatrix} \omega_W \\ \dot{p}_W \end{bmatrix} = J(\theta)\dot{\theta} \quad (121)$$

Con ω_W como velocidad angular de la muñeca y \dot{p}_W velocidad lineal de la muñeca.

La velocidad de la muñeca v_W puede relacionarse con la velocidad del efector final v_E compuesto por los vectores ω_E y \dot{p}_E a partir de la siguiente fórmula:

$$v_W = \begin{bmatrix} 1 & O \\ P - P_W & 1 \end{bmatrix} v_E \quad (122)$$

Con P y P_W definidas como las matrices antisimétricas de los vectores p y p_W respectivamente.

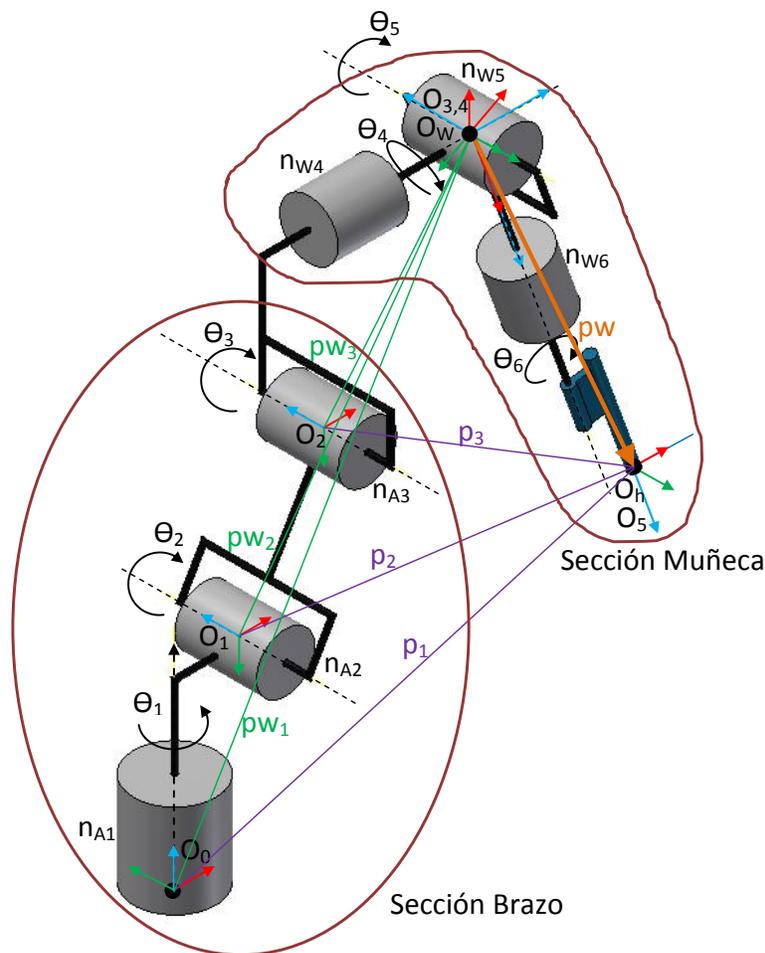


Figura 17 Sistema coordenadas cinemática diferencial y manipulabilidad

Dado que O_w se encuentra en la intersección de los ejes de las tres últimas articulaciones la velocidad lineal de la muñeca v_w no es afectada por el movimiento de las tres últimas articulaciones, por consiguiente:

$$\dot{p}_w = \dot{\theta}_1 {}^0 z_0 \times {}^1 p_{w_1} + \dot{\theta}_2 {}^0 z_1 \times {}^1 p_{w_2} + \dot{\theta}_3 {}^0 z_2 \times {}^1 p_{w_3} \quad (123)$$

Donde ${}^1 p_{w_i}$, para el caso de un manipulador desacoplado, es el vector dirigido desde O_i hasta O_w , y ${}^0 z_{i-1}$, denota el vector unidad en la dirección del eje Z de la articulación i respecto al marco de referencia cero. Cabe anotar que:

$${}^0 z_{i-1} = \prod_{k=2}^i ({}^{k-2} R_{k-1}) {}^0 z_0 \quad (124)$$

Donde i , se encuentra en el rango de 2 a 6, y ${}^0 z_0$ es igual a $[0 \ 0 \ 1]^T$, y ${}^{k-2} R_{k-1}$ es la matriz de rotación de la articulación respecto a su marco de referencia O_i .

En la expresión:

$${}^1 p_{w_i} = \sum_{k=i}^4 {}^1 P_k \quad (125)$$

i , se encuentra en el rango de 1 a 3, y ${}^1 P_k$ es el vector de posición de la articulación i respecto al marco de referencia de la base (O_0). Por consiguiente:

$${}^1 P_k = \prod_{m=2}^k ({}^{m-2} R_{m-1}) {}^m P_m \quad (126)$$

Donde k , se encuentra en el rango de 2 a 4, y ${}^m P_m$ es el vector de posición de la articulación i respecto al marco de referencia O_i .

Por otra parte se tiene que:

$$\omega_w = \dot{\theta}_1 {}^0 z_0 + \dot{\theta}_2 {}^0 z_1 + \dot{\theta}_3 {}^0 z_2 + \dot{\theta}_4 {}^0 z_3 + \dot{\theta}_5 {}^0 z_4 + \dot{\theta}_6 {}^0 z_5 \quad (127)$$

Y por lo tanto la matriz Jacobiana puede adquirir la siguiente forma simplificada:

$$J = \begin{bmatrix} J_{RA} & J_{RW} \\ J_{TA} & O \end{bmatrix} \quad (128)$$

Con O equivalente a una matriz cero de tamaño (3x3), y formulando las tres expresiones restantes como:

$$J_{RA} = \begin{bmatrix} {}^0 z_0 & {}^0 z_1 & {}^0 z_2 \end{bmatrix} \quad (129)$$

$$J_{RW} = \begin{bmatrix} {}^0z_3 & {}^0z_4 & {}^0z_5 \end{bmatrix} \quad (130)$$

$$J_{TA} = \begin{bmatrix} {}^0z_0 \times^1 p w_1 & {}^0z_1 \times^1 p w_2 & {}^0z_2 \times^1 p w_3 \end{bmatrix} \quad (131)$$

Donde J_{RA} representa la Jacobiana de la parte rotacional de la sección del brazo, J_{RW} representa la Jacobiana de la parte rotacional de la sección de la muñeca y J_{TA} representa la Jacobiana de la parte traslacional de la sección del brazo.

Además el vector de velocidades articulares puede dividirse en:

$$\dot{\theta} = \begin{bmatrix} \theta_A \\ \theta_W \end{bmatrix} \quad (132)$$

Donde θ_A es el vector de velocidades articulares de la sección del brazo y θ_W es el vector de velocidades articulares de la sección de la muñeca. Los cuales a su vez son iguales a:

$$\dot{\theta}_A = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}, \quad \dot{\theta}_W = \begin{bmatrix} \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \quad (133)$$

Relacionando las velocidades angulares y lineales de la muñeca (ω_W y $\dot{p}w$ respectivamente) con las matrices Jacobianas se tiene que:

$$\omega_W = J_{RA} \dot{\theta}_A + J_{RW} \dot{\theta}_W \quad (134)$$

$$\dot{p}w = J_{TA} \dot{\theta}_A \quad (135)$$

Partiendo de las ecuaciones anteriores es posible hallar las velocidades articulares, conociendo las velocidades angulares y lineales de la muñeca, encontradas previamente con la ecuación 122:

$$\dot{\theta}_A = J_{TA}^{-1} \dot{p}w \quad (136)$$

$$\dot{\theta}_W = J_{RW}^{-1} (\omega_W - J_{RA} \dot{\theta}_A) \quad (137)$$

Aplicando el concepto de bases recíprocas para hallar la inversa de una matriz de (3x3), se tiene que:

$$J_{TA}^{-1} = \frac{1}{\Delta_{TA}} \begin{bmatrix} \left[\begin{matrix} ({}^0z_1 \times^1 p w_2) \times ({}^0z_2 \times^1 p w_3) \end{matrix} \right]^T \\ \left[\begin{matrix} ({}^0z_2 \times^1 p w_3) \times ({}^0z_0 \times^1 p w_1) \end{matrix} \right]^T \\ \left[\begin{matrix} ({}^0z_0 \times^1 p w_1) \times ({}^0z_1 \times^1 p w_2) \end{matrix} \right]^T \end{bmatrix} \quad (138)$$

Con:

$$\Delta_{TA} = \det(J_{TA}) = ({}^0z_0 \times^1 p w_1) \times ({}^0z_1 \times^1 p w_2) \bullet ({}^0z_2 \times^1 p w_3) \quad (139)$$

Y,

$$J_{RW}^{-1} = \frac{1}{\Delta_{RW}} \begin{bmatrix} \left({}^0z_4 \times {}^0z_5 \right)^T \\ \left({}^0z_5 \times {}^0z_3 \right)^T \\ \left({}^0z_3 \times {}^0z_4 \right)^T \end{bmatrix} \quad (140)$$

Con:

$$\Delta_{RW} = \det(J_{RW}) = {}^0z_3 \times {}^0z_4 \bullet {}^0z_6 \quad (141)$$

Por consiguiente:

$$\dot{\theta}_A = \frac{1}{\Delta_{TA}} \begin{bmatrix} \left({}^0z_1 \times {}^1pw_2 \right) \times \left({}^0z_2 \times {}^1pw_3 \right) \bullet \dot{pw} \\ \left({}^0z_2 \times {}^1pw_3 \right) \times \left({}^0z_0 \times {}^1pw_1 \right) \bullet \dot{pw} \\ \left({}^0z_0 \times {}^1pw_1 \right) \times \left({}^0z_1 \times {}^1pw_2 \right) \bullet \dot{pw} \end{bmatrix} \quad (142)$$

$$\dot{\omega}_W = \frac{1}{\Delta_{RW}} \begin{bmatrix} \left({}^0z_4 \times {}^0z_5 \right) \bullet (\omega_W - J_{RA} \dot{\theta}_A) \\ \left({}^0z_5 \times {}^0z_3 \right) \bullet (\omega_W - J_{RA} \dot{\theta}_A) \\ \left({}^0z_3 \times {}^0z_4 \right) \bullet (\omega_W - J_{RA} \dot{\theta}_A) \end{bmatrix} \quad (143)$$

2.3.2 CÁLCULO ACELERACIONES ARTICULARES

Partiendo de la ecuación 135 de la velocidad lineal de la muñeca \dot{pw} , la aceleración lineal \ddot{pw} de la misma puede ser expresada de la siguiente forma:

$$\ddot{pw} = \dot{J}_{TA} \dot{\theta}_A + J_{TA} \ddot{\theta}_A \quad (144)$$

De la expresión anterior, la ecuación del vector de las aceleraciones articulares de la sección del brazo queda finalmente expresada como:

$$\ddot{\theta}_A = J_{TA}^{-1} (\ddot{pw} - \dot{J}_{TA} \dot{\theta}_A) \quad (145)$$

Partiendo de la ecuación 134 de la velocidad angular de la muñeca ω_W , la aceleración angular $\dot{\omega}_W$ de la misma puede ser expresada de la siguiente forma:

$$\dot{\omega}_W = \dot{J}_{RA} \dot{\theta}_A + J_{RA} \ddot{\theta}_A + \dot{J}_{RW} \dot{\omega}_W + J_{RW} \ddot{\omega}_W \quad (146)$$

De la expresión anterior la ecuación del vector de las aceleraciones articulares de la sección de la muñeca queda finalmente expresada como:

$$\ddot{\omega}_W = J_{RW}^{-1} (\dot{\omega}_W - \dot{J}_{RA} \dot{\theta}_A - J_{RA} \ddot{\theta}_A - \dot{J}_{RW} \dot{\omega}_W) \quad (147)$$

Las derivadas de las Jacobianas se expresan como:

$$J_{RA} = \begin{bmatrix} {}^1\omega_1 \times^0 z_0 & {}^1\omega_2 \times^0 z_1 & {}^1\omega_3 \times^0 z_2 \end{bmatrix} \quad (148)$$

$$J_{RW} = \begin{bmatrix} {}^1\omega_4 \times^0 z_3 & {}^1\omega_5 \times^0 z_4 & {}^1\omega_6 \times^0 z_5 \end{bmatrix} \quad (149)$$

$$J_{TA} = \left[\left({}^1\omega_1 \times^0 z_0 \right) \times^1 p w_1 + {}^0 z_0 \times^1 \dot{p} w_1 \quad \left({}^1\omega_2 \times^0 z_1 \right) \times^1 p w_2 + {}^0 z_1 \times^1 \dot{p} w_2 \quad \left({}^1\omega_3 \times^0 z_2 \right) \times^1 p w_3 + {}^0 z_2 \times^1 \dot{p} w_3 \right] \quad (150)$$

Dónde:

$${}^1\omega_i = \sum_{k=1}^i {}^o z_{k-1} \theta_k \quad (151)$$

Con i comprendida en el rango de 1 a 6.

Y:

$${}^1 \dot{p} w_i = \sum_{k=i}^4 \left({}^1\omega_k + {}^1 P_k \right) \quad (152)$$

Con i comprendida en el rango de 1 a 3.

2.3.3 MANIPULABILIDAD TRANSLACIONAL Y ROTACIONAL MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

La medida de la manipulabilidad translacional w_A y la medida de la manipulabilidad rotacional w_W para un manipulador antropomórfico de seis grados de libertad se hallan aplicando las ecuaciones 35 y 36 de manipulabilidad, halladas en la sección 1.3.3, a los parámetros cinemáticos del manipulador. Desarrollando estas ecuaciones las expresiones simplificadas de las medidas de las manipulabilidades son:

Manipulabilidad translacional o de brazo

$$w_A = a_2 \left[d_4 \text{Cos}(\theta_3) + a_3 \text{Sen}(\theta_3) \right] \cdot \left[a_1 + a_3 \text{Cos}(\theta_2 + \theta_3) - d_4 \text{Sen}(\theta_2 + \theta_3) + a_2 \text{Cos}(\theta_2) \right] \quad (153)$$

Manipulabilidad rotacional o de muñeca

$$w_W = \text{Sen}(\theta_5) \quad (154)$$

2.4 DINÁMICA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

2.4.1 DINÁMICA INVERSA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

Para hallar la dinámica inversa del manipulador antropomórfico de seis grados de libertad y muñeca esférica se opta por implementar el algoritmo recursivo de Newton-Euler definiendo el rango de i entre 1 y 6 que es el equivalente a las seis articulaciones del manipulador, del cual se tienen las siguientes expresiones:

Velocidad angular eslabón i :

$${}^i\omega_i = {}^{i-1}R_i^T ({}^{i-1}\omega_{i-1} + \dot{\theta}_i {}^0z_0) \quad (155)$$

Aceleración angular eslabón i :

$${}^i\dot{\omega}_i = {}^{i-1}R_i^T \left[{}^{i-1}\dot{\omega}_{i-1} + \ddot{\theta}_i {}^0z_0 + {}^{i-1}\omega_{i-1} \times (\dot{\theta}_i {}^0z_0) \right] \quad (156)$$

Aceleración lineal eslabón i :

$${}^i\dot{v}_i = {}^{i-1}R_i^T * {}^{i-1}\dot{v}_{i-1} + {}^i\dot{\omega}_i \times {}^i p_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i p_i) \quad (157)$$

Aceleración del centro de masa del eslabón i :

$${}^i A_{CMi} = {}^i\dot{v}_i + {}^i\dot{\omega}_i \times {}^i s_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i s_i) \quad (158)$$

Fuerza ejercida por el eslabón $i-1$ en el eslabón i :

$${}^i F_i = {}^i R_{i+1} * {}^{i+1} F_{i+1} + m_i {}^i A_{CMi} \quad (159)$$

Momento ejercido por el eslabón $i-1$ en el eslabón i . respecto al origen del Marco de referencia $i-1$.

$${}^i P_{ARi} = {}^i R_{i+1} * {}^{i+1} P_{ARi+1} + {}^i R_{i+1} * {}^{i+1} F_{i+1} \times {}^i p_i - {}^i F_i \times ({}^i p_i + {}^i s_i) + {}^i I_i \dot{\omega}_i + {}^i \omega_i \times ({}^i I_i \omega_i) \quad (160)$$

Par/ fuerza generalizado en la articulación i se expresa por:

$$\tau_i = {}^i R_{i+1} * {}^i P_{ARi} {}^T 0 z_0 + b_i \dot{\theta}_i \quad (161)$$

Las ecuaciones anteriores poseen una gran ventaja debido a que las cantidades ${}^i I_i$, ${}^i S_i$, son constantes y como se mencionó anteriormente ${}^0 z_0$ es igual a $[0 \ 0 \ 1]^T$.

2.4.2 CONSUMO DE ENERGÍA MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

Una vez se identifica la dinámica inversa del manipulador y considerando la fuente de movimiento como motores de corriente directa de imán permanente, las ecuaciones necesarias para hallar el consumo final del manipulador antropomórfico de seis grados de libertad y muñeca esférica son:

Ecuación de corriente I arrastrada por los motores de cada articulación i del manipulador:

$$I_i = \frac{\tau_{mi}}{K_{ti}} \quad (162)$$

Ecuación del potencial electromotriz Ve de los motores de cada articulación i del manipulador:

$$Ve_i = K_{ei} \omega_{mi} \quad (163)$$

Donde ω_{mi} es la velocidad angular de los actuadores K_{ti} es el factor de sensibilidad del motor o constante de motor expresada en (Nm/A), K_{ei} la constante de fuerza contra electromotriz expresada en (V*(rad/seg)⁻¹) y τ_{mi} es el torque necesario en los motores, el cual es igual a:

$$\tau_{mi} = \frac{\tau_i}{n_{ri}} \quad (164)$$

Con n_{ri} representando la relación de transmisión de los reductores que acoplan la articulación i con los motores.

Ecuación de pérdida de potencia resistiva (efecto Joule):

$$P_{Ji} = R_i I_i^2 \quad (165)$$

Ecuación de potencia usada para producir la fuerza electromotriz:

$$P_{EMi} = Ve_i I_i \quad (166)$$

Donde R es la resistencia de la armadura del motor expresada en Ohm.

Por consiguiente, la potencia eléctrica total P_T consumida por los seis motores puede expresarse como:

$$P_T = \sum_{i=1}^6 P_{Ji} + P_{EMi} \quad (167)$$

Finalmente, el consumo eléctrico C consumido por el manipulador a lo largo de una trayectoria se evalúa mediante una sumatoria de la energía consumida de los seis motores por el tiempo en horas en cada punto de la trayectoria de n puntos:

$$C = \sum_{k=1}^n P_{Tk} t_k \text{ (KWH)} \quad (168)$$

P_{Tk} hace referencia a la potencia eléctrica instantánea consumida en un punto k por los seis motores del manipulador en un instante de tiempo t_k .

2.5 ESPACIO DE TRABAJO SIMPLIFICADO EN MANIPULADORES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD CON MUÑECA ESFÉRICA

El objeto de este apartado es plantear las ecuaciones que permitan encontrar las coordenadas cartesianas de un punto dentro del espacio de trabajo simplificado propio para el manipulador antropomórfico de seis grados de libertad con muñeca esférica. Esto será el punto de partida para la generación del algoritmo de planeación de trayectorias.

El espacio de trabajo del manipulador será tomado como un espacio esférico, ubicando su origen en el punto final de la distancia d_1 (O_s) propia del manipulador, y estará delimitado por un radio mayor (RM_y) y menor (RM_n) los cuales comprenden la distancia hasta la muñeca del manipulador (O_h), un ángulo de colatitud superior (γ_{RS}) e inferior (γ_{RI}) y un ángulo azimutal superior (ϕ_{RS}) e inferior (ϕ_{RI}) Fig. 18. Un espacio de trabajo delimitado de esta forma permite reducir el número de variables que lo definen y restringir el espacio a zonas donde la ubicación y orientación del efector final cumple con los requerimientos de orientación necesarios para obtener buenas piezas en el proceso de soldadura de arco para lámina metálica. Los requerimientos de orientación del efector final en el proceso de soldadura horizontal son ángulos de viaje de 65 a 80° respecto a la horizontal, y en soldaduras verticales, ángulos de viaje de 15 a 20° respecto a la horizontal, cuando la soldadura es descendente, y ángulos de viaje de 0 a 5° cuando la soldadura es ascendente. (Saunders, 1997)

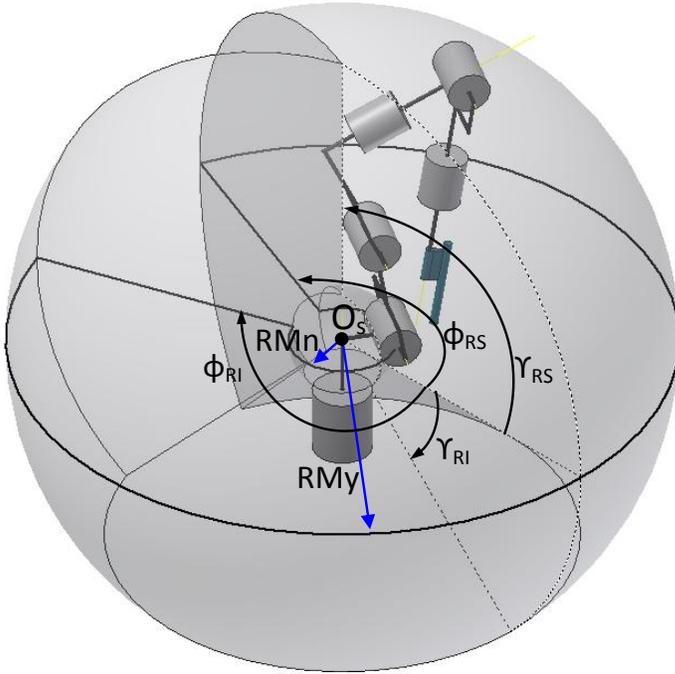


Figura 18 Espacio de trabajo simplificado manipulador antropomórfico seis grados de libertad y muñeca esférica

Para comenzar con el cálculo, se debe contar con los parámetros cinemáticos del manipulador, y los ángulos límite de sus articulaciones dados por el fabricante del manipulador. Posteriormente mediante un procedimiento geométrico, se hallan los valores máximos y mínimos de las variables del espacio de trabajo y se plantean las ecuaciones de las coordenadas finales del manipulador en función de estas variables. El desarrollo de este procedimiento se expone a continuación (Fig. 19):

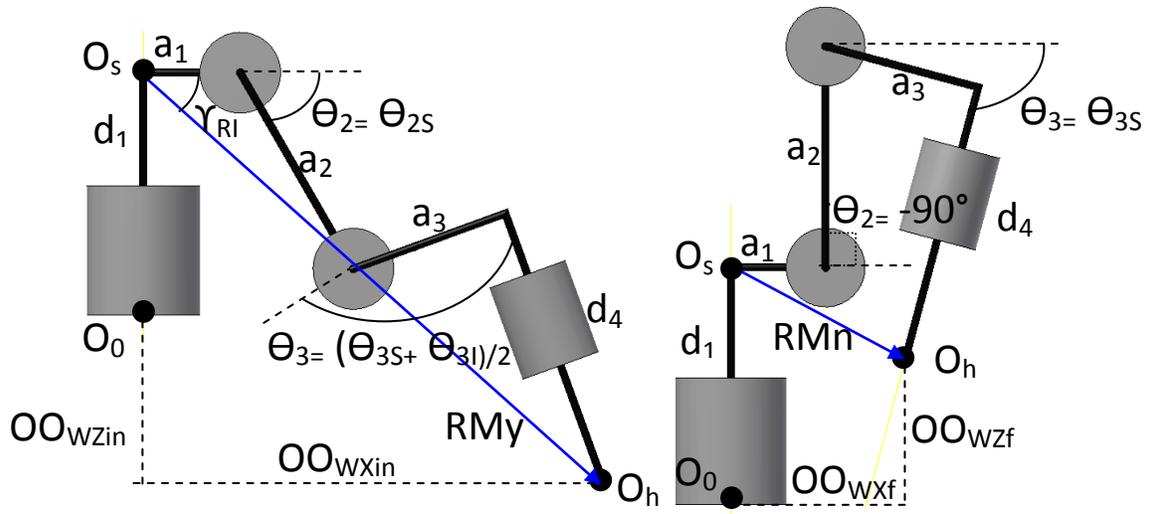


Figura 19 Esquema para cálculo de radio mayor y menor de espacio de trabajo manipulador antropomórfico seis grados de libertad y muñeca esférica

Partiendo de los siguientes datos para las articulaciones del manipulador:

$$\theta_1 = 0^\circ$$

$\theta_2 = \theta_{2S}$, donde θ_{2S} es límite superior de la articulación 2 del manipulador.

$\theta_3 = (\theta_{3S} + \theta_{3I})$, donde θ_{3S} es límite superior de la articulación 3 del manipulador y θ_{3I} es límite inferior de la articulación 3 del manipulador.

$$\theta_4 = 180^\circ, \theta_5 = 0^\circ \text{ y } \theta_6 = 0^\circ$$

Además, con base al esquema del manipulador de la Fig. 19, y hallando las coordenadas (OO_{wxin} , OO_{wyin} , OO_{wzin}) del origen de la muñeca (O_h) del manipulador, según el procedimiento descrito en la sección de la cinemática inversa (ecuación 100), se calcula la ecuación del radio mayor del espacio de trabajo (RM_y):

$$RM_y = \sqrt{(OO_{wxin})^2 + (OO_{wzin} - d_1)^2} \quad (169)$$

El radio menor (RM_n) se calcula de igual forma, pero se toman los siguientes datos para las articulaciones del manipulador:

$$\theta_1 = 0^\circ$$

$\theta_2 = -90^\circ$, debido a que es el ángulo en el cual la articulación 3 se encontrará más alejada verticalmente de la base.

$\theta_3 = \theta_{3S}$, donde θ_{3S} es límite superior de la articulación 3 del manipulador.

$$\theta_4 = 180^\circ, \theta_5 = 0^\circ \text{ y } \theta_6 = 0^\circ$$

Por consiguiente la ecuación del radio menor es:

$$RM_n = \sqrt{(OO_{wxf})^2 + (OO_{wzf} - d_1)^2} \quad (170)$$

Con el radio mayor (RM_y) se procede con el cálculo del ángulo inferior de colatitud (γ_{RI}):

$$\gamma_{RI} = ACos\left(\frac{OO_{wxin}}{RM_y}\right) \quad (171)$$

Y se fija el valor del ángulo superior de colatitud (γ_{RS}) en -90° , abarcando con esto el cuadrante superior de la esfera del espacio de trabajo.

Los ángulos azimutal límites del espacio de trabajo corresponden con los ángulos límites inferior y superior de la articulación 1, por consiguiente:

$$\phi_{RI} = \theta_{1I} \quad (172)$$

$$\phi_{RS} = \theta_{1S} \quad (173)$$

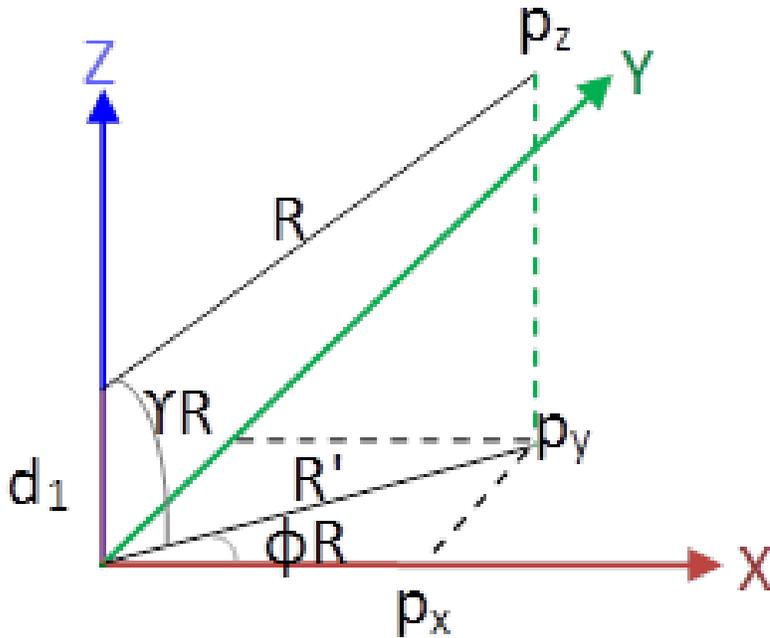


Figura 20 Esquema coordenadas cartesianas a partir de variables de espacio de trabajo simplificado

Por último la ecuación de las coordenadas cartesianas de un punto dentro del espacio de trabajo simplificado según las variables halladas y el esquema de la Fig. 20, son:

$$p_x = Cos(\phi)|Cos(\gamma)R| \quad (174)$$

$$p_y = \text{Sen}(\phi) |\text{Cos}(\gamma) R| \quad (175)$$

$$p_z = -\text{Sen}(\gamma) R + d_1 \quad (176)$$

2.6 MÉTODO OPTIMIZACIÓN HKA APLICADO A LOS CRITERIOS DE CONSUMO ELÉCTRICO Y MANIPULABILIDAD DEL MANIPULADOR ANTROPOMÓRFICO DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA

El algoritmo de optimización propuesto tiene como finalidad minimizar el criterio de consumo eléctrico de la trayectoria y maximizar la manipulabilidad promedio de la trayectoria (w) (o minimizar $1-w$), partiendo de la variación de la ubicación de la trayectoria por medio de las variables radio del espacio de trabajo (R), ángulo de colatitud (γ), y ángulo azimutal (ϕ), dentro de los límites del espacio de trabajo simplificado.

La función objetivo o de costo a minimizar $J(q)$, está compuesta por dos criterios: la minimización del consumo eléctrico de la trayectoria del manipulador y la maximización de la manipulabilidad de la trayectoria (o minimización de $1-w$). Esta función de costo puede expresarse como:

$$J(q) = \frac{C}{Pn * t} * PC + \frac{(1-w_p) - (1-w_M)}{1 - (1-w_M)} * PM \quad (177)$$

Dónde:

C : Consumo de la trayectoria en KWH

Pn : Potencia nominal del manipulador en Kw.

t : Tiempo de la trayectoria en horas.

PC : Peso asignado a criterio consumo eléctrico de la trayectoria (de 0 a 1).

w_p : manipulabilidad promedio de la trayectoria.

w_M : manipulabilidad máxima en el manipulador.

$PM = 1 - PC$: Peso asignado a criterio de la manipulabilidad de la trayectoria (de 0 a 1).

Esta función de costo $J(q)$ es dependiente de las variables de ubicación inicial de la trayectoria, conocidas como variables de decisión: radio del espacio de trabajo (R), ángulo de colatitud (γ), y ángulo azimutal (ϕ), las cuales se presentan a continuación:

\underline{q}_i	\leq	q	\leq	\bar{q}_i
RMn	\leq	R	\leq	RM_y
γ_{RI}	\leq	γ	\leq	γ_{RS}
ϕ_{RI}	\leq	ϕ	\leq	ϕ_{RS}

Tabla 3 Variables de decisión algoritmo HKA

Debido a que el sistema está sometido a una serie de restricciones, se debe replantear la ecuación de la función de costo $J(q)$ teniendo en cuenta dichas restricciones, como se expone a continuación:

$$J_n(q) = J(q) + \varpi \sum_{i=1}^{N_c} \max(g_i(q), 0) \quad (178)$$

Dónde:

$J_n(q)$: Nueva función de costo

ϖ : factor de peso para las restricción. Se tomará como 100.

$g_i(q)$: Función de la restricción i -ésima.

N_c : Número de restricciones. Para este sistema se definieron 13 restricciones las cuales se indican a continuación:

$$g_1(q) = \sum_{p=1}^{N_p} (\theta_{1l} < \theta_{1p} < \theta_{1s} \rightarrow S\theta_{1p} = 1) \wedge (\theta_{1s} \leq \theta_{1p} \leq \theta_{1l} \rightarrow S\theta_{1p} = 0) \quad (179)$$

$$g_2(q) = \sum_{p=1}^{N_p} (\theta_{2l} < \theta_{2p} < \theta_{2s} \rightarrow S\theta_{2p} = 1) \wedge (\theta_{2s} \leq \theta_{2p} \leq \theta_{2l} \rightarrow S\theta_{2p} = 0) \quad (180)$$

$$g_3(q) = \sum_{p=1}^{Np} (\theta_{3I} < \theta_{3p} < \theta_{3S} \rightarrow S\theta_{3p} = 1) \wedge (\theta_{3S} \leq \theta_{3p} \leq \theta_{3I} \rightarrow S\theta_{3p} = 0) \quad (181)$$

$$g_4(q) = \sum_{p=1}^{Np} (\theta_{4I} < \theta_{4p} < \theta_{4S} \rightarrow S\theta_{4p} = 1) \wedge (\theta_{4S} \leq \theta_{4p} \leq \theta_{4I} \rightarrow S\theta_{4p} = 0) \quad (182)$$

$$g_5(q) = \sum_{p=1}^{Np} (\theta_{5I} < \theta_{5p} < \theta_{5S} \rightarrow S\theta_{5p} = 1) \wedge (\theta_{5S} \leq \theta_{5p} \leq \theta_{5I} \rightarrow S\theta_{5p} = 0) \quad (183)$$

$$g_6(q) = \sum_{p=1}^{Np} (\theta_{6I} < \theta_{6p} < \theta_{6S} \rightarrow S\theta_{6p} = 1) \wedge (\theta_{6S} \leq \theta_{6p} \leq \theta_{6I} \rightarrow S\theta_{6p} = 0) \quad (184)$$

$$g_7(q) = \sum_{p=1}^{Np} (\dot{\theta}_{1I} < \dot{\theta}_{1p} < \dot{\theta}_{1S} \rightarrow S\dot{\theta}_{1p} = 1) \wedge (\dot{\theta}_{1S} \leq \dot{\theta}_{1p} \leq \dot{\theta}_{1I} \rightarrow S\dot{\theta}_{1p} = 0) \quad (185)$$

$$g_8(q) = \sum_{p=1}^{Np} (\dot{\theta}_{2I} < \dot{\theta}_{2p} < \dot{\theta}_{2S} \rightarrow S\dot{\theta}_{2p} = 1) \wedge (\dot{\theta}_{2S} \leq \dot{\theta}_{2p} \leq \dot{\theta}_{2I} \rightarrow S\dot{\theta}_{2p} = 0) \quad (186)$$

$$g_9(q) = \sum_{p=1}^{Np} (\dot{\theta}_{3I} < \dot{\theta}_{3p} < \dot{\theta}_{3S} \rightarrow S\dot{\theta}_{3p} = 1) \wedge (\dot{\theta}_{3S} \leq \dot{\theta}_{3p} \leq \dot{\theta}_{3I} \rightarrow S\dot{\theta}_{3p} = 0) \quad (187)$$

$$g_{10}(q) = \sum_{p=1}^{Np} (\dot{\theta}_{4I} < \dot{\theta}_{4p} < \dot{\theta}_{4S} \rightarrow S\dot{\theta}_{4p} = 1) \wedge (\dot{\theta}_{4S} \leq \dot{\theta}_{4p} \leq \dot{\theta}_{4I} \rightarrow S\dot{\theta}_{4p} = 0) \quad (188)$$

$$g_{11}(q) = \sum_{p=1}^{Np} (\dot{\theta}_{5I} < \dot{\theta}_{5p} < \dot{\theta}_{5S} \rightarrow S\dot{\theta}_{5p} = 1) \wedge (\dot{\theta}_{5S} \leq \dot{\theta}_{5p} \leq \dot{\theta}_{5I} \rightarrow S\dot{\theta}_{5p} = 0) \quad (189)$$

$$g_{12}(q) = \sum_{p=1}^{Np} (\dot{\theta}_{6I} < \dot{\theta}_{6p} < \dot{\theta}_{6S} \rightarrow S\dot{\theta}_{6p} = 1) \wedge (\dot{\theta}_{6S} \leq \dot{\theta}_{6p} \leq \dot{\theta}_{6I} \rightarrow S\dot{\theta}_{6p} = 0) \quad (190)$$

$$g_{13}(q) = \sum_{p=1}^{Np} (Pm_p > Pn \rightarrow SPm_p = 1) \wedge (Pm_p \leq Pn \rightarrow SPm_p = 0) \quad (191)$$

En estas:

Np : Indica el número de puntos de la trayectoria.

$\theta_{1I}, \theta_{2I} \dots \theta_{6I}$: Límite inferior de las seis articulaciones del robot.

$\theta_{1S}, \theta_{2S} \dots \theta_{6S}$: Límite superior de las seis articulaciones del robot.

$S\theta_{1p}, S\theta_{2p} \dots S\theta_{6p}$: Variable que indica cuando la articulación correspondiente del robot (de 1 a 6) en el punto p de la trayectoria esta fuera de sus límites, siendo 1 cuando está fuera de los límites y cero cuando cumple con estos.

$\dot{\theta}_{1I}, \dot{\theta}_{2I} \dots \dot{\theta}_{6I}$: Límite inferior de las velocidades de las seis articulaciones del robot.

$\dot{\theta}_{1S}, \dot{\theta}_{2S} \dots \dot{\theta}_{6S}$: Límite superior de las velocidades de las seis articulaciones del robot.

$S\dot{\theta}_{1p}, S\dot{\theta}_{2p} \dots S\dot{\theta}_{6p}$: Variable que indica cuando la velocidad de la articulación correspondiente del robot (de 1 a 6) en el punto p de la trayectoria esta fuera de sus límites, siendo 1 cuando está fuera de los límites y cero cuando cumple con estos.

Pm_p : Variable que almacena la suma de la potencia de los seis motores (Kw) del manipulador en el punto p de la trayectoria.

Pn : Potencia nominal del manipulador en Kw.

SPm_p : Variable que indica cuando la potencia de los seis motores en el punto p de la trayectoria supera la potencia nominal del manipulador, siendo 1 cuando la supera y cero cuando no lo hace.

Los parámetros de inicialización del generador Gaussiano, se hallan a partir de las variables de decisión del algoritmo HKA, mediante la ecuación 72:

$$\mu_1 = \frac{RM_y + RM_n}{2} \quad (192)$$

$$\mu_2 = \frac{\gamma_{RS} + \gamma_{RI}}{2} \quad (193)$$

$$\mu_3 = \frac{\phi_{RS} + \phi_{RI}}{2} \quad (194)$$

$$m_0 = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} \quad (195)$$

$$\sigma_1 = \frac{RM_y - RM_n}{6} \quad (196)$$

$$\sigma_2 = \frac{\gamma_{RS} - \gamma_{RI}}{6} \quad (197)$$

$$\sigma_3 = \frac{\phi_{RS} - \phi_{RI}}{6} \quad (198)$$

$$\Sigma_0 = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} \quad (199)$$

La configuración del algoritmo se realiza con N igual a 50 muestras (o trayectorias), y N_ξ igual a 5 mejores muestras (o trayectorias). Esto se corrobora al aplicar la ecuación 74, debido a que se obtiene una probabilidad mayor a 0.95:

$$p_B^5 \geq \frac{1}{2^{50}} \sum_{i=5}^{50} \frac{50!}{i!(50-i)!} \text{ donde } p_B^5 \geq 1 \quad (200)$$

El coeficiente de ralentización α del algoritmo se concretará con el algoritmo implementado. Se le dará inicialmente un valor de 0.3.

La regla de paro del algoritmo se configura de forma tal que en 10 iteraciones seguidas se debe cumplir con las siguientes condiciones:

El valor absoluto de la diferencia entre $1-w$ de la iteración actual y la de la iteración anterior debe ser menor a 0.0005.

La función de costo de la iteración actual debe ser menor a 0.9

El valor absoluto de la diferencia entre el consumo eléctrico de la iteración actual y la de la iteración anterior debe ser menor a 0.0000001.

El valor absoluto de la diferencia entre el porcentaje de consumo eléctrico de la iteración actual y la de la iteración anterior debe ser menor a 0.0005.

Una vez se tienen identificadas los parámetros de inicialización del generador Gaussiano, las variables de configuración del algoritmo HKA, y la regla de paro del algoritmo se procede a desarrollar el algoritmo de acuerdo a los pasos planteados en el numeral 1.5.2 (Desarrollo del Algoritmo HKA).

2.7 ALGORITMO DE PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD

En esta sección se exponen las tablas de variables de entrada y las tablas de variables de salida del algoritmo de optimización de trayectorias en manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica basado en HKA propuesto, el cual tiene como objetivo minimizar el consumo eléctrico y maximizar la manipulabilidad.

2.7.1 VARIABLES DE ENTRADA

A continuación se presenta un listado de las variables de entrada a considerar en cada manipulador serial antropomórfico de seis grados de libertad y muñeca esférica, necesarias para llevar a cabo el proceso de optimización de trayectorias basado en manipulabilidad y consumo eléctrico.

PARÁMETROS CINEMÁTICOS			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Separación del eslabón 1	d_1	m	Variable Real
Longitud del eslabón 1	a_1	m	Variable Real
Orientación del eslabón 1	alfa_1	°	Variable Real
Longitud del eslabón 2	a_2	m	Variable Real
Separación del eslabón 3	d_3	m	Variable Real
Longitud del eslabón 3	a_3	m	Variable Real
Orientación del eslabón 3	alfa_3	°	Variable Real
Separación del eslabón 4	d_4	m	Variable Real
Orientación del eslabón 4	alfa_4	°	Variable Real
Orientación del eslabón 5	alfa_5	°	Variable Real
Separación del eslabón 6	d_6	m	Variable Real
Separación del efector final	d_h	m	Variable Real
Longitud del efector final	a_h	m	Variable Real
Matriz (6x2) límite desplazamiento inferior y superior articulaciones	L_ART()	°	Matriz de variables reales
Vector (6x1) límite de velocidades articulaciones	L_VART()	°/s	Matriz de variables reales

Tabla 4 Variables entrada parámetros cinemáticos manipuladores antropomórficos de seis grados de libertad y muñeca esférica

PARÁMETROS DIFERENCIALES			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Velocidad lineal máxima efector final	VelS ()	m/s	Variable Real
Velocidad angular X efector final	WX	rad/s	Variable Real
Velocidad angular Y efector final	WY	rad/s	Variable Real
Velocidad angular Z efector final	WZ	rad/s	Variable Real
Aceleración angular X efector final	AcWX	rad/s ²	Variable Real
Aceleración angular Y efector final	AcWY	rad/s ²	Variable Real
Aceleración angular Z efector final	AcWZ	rad/s ²	Variable Real
Aceleración lineal X efector final	AcX	m/s ²	Variable Real
Aceleración lineal Y efector final	AcY	m/s ²	Variable Real
Aceleración lineal Z efector final	AcZ	m/s ²	Variable Real

Tabla 5 Variables entrada parámetros diferenciales manipuladores antropomórficos de seis grados de libertad y muñeca esférica.

PARÁMETROS DINÁMICOS			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Vector (7x1) peso de los eslabones y efector final	Mm()	Kg	Vector de variables reales
Matriz (7x3) centro de masa XYZ eslabones robot y efector final	M_s ()	m	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 1	In1()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 2	In2()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 3	In3()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 4	In4()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 5	In5()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia eslabón 6	In6()	Kg*m ²	Matriz de variables reales
Matriz (3x3) momento de inercia efector final o herramienta	IPist()	Kg*m ²	Matriz de variables reales
Vector (6x1) coeficientes viscosidad motores robot	b_()	N.m.s/rad	Vector de variables reales
Vector (6x1) constante torque motores robot	Kt()	N.m/A	Vector de variables reales
Vector (6x1) constante de fuerza contra electromotriz motores robot	Ke()	V.s/rad	Vector de variables reales
Vector (6x1) resistencia armaduras motores robot	R_()	Ohms	Vector de variables reales
Vector (6x1) relación de transmisión motores robot	nr()	----	Vector de variables reales

Potencia nominal robot	Pn	Kw	Variable Real
------------------------	----	----	---------------

Tabla 6 Variables entrada parámetros dinámicos manipuladores antropomórficos de seis grados de libertad y muñeca esférica.

TRAYECTORIA			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Matriz (nx7) trayectoria a optimizar	$T_{\text{Análisis}}()$	Varias	Matriz de variables reales
Dónde i es el punto i-ésimo de la trayectoria de n puntos.			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Punto i-ésimo de la trayectoria	$T_{\text{Análisis}}(i, 1)$	----	Variable Real Entera
Coordenada X efector final (pX ₋)	$T_{\text{Análisis}}(i, 2)$	m	Variable Real
Coordenada Y efector final (pY ₋)	$T_{\text{Análisis}}(i, 3)$	m	Variable Real
Coordenada Z efector final (pZ ₋)	$T_{\text{Análisis}}(i, 4)$	m	Variable Real
Angulo de Euler de precesión para orientación efector final (APrec)	$T_{\text{Análisis}}(i, 5)$	rad	Variable Real
Angulo de Euler de nutación para orientación efector final (ANut)	$T_{\text{Análisis}}(i, 6)$	rad	Variable Real
Angulo de Euler de rotación propia para orientación efector final (ARotP)	$T_{\text{Análisis}}(i, 7)$	rad	Variable Real

Tabla 7 Variables entrada trayectoria manipuladores antropomórficos de seis grados de libertad y muñeca esférica.

Las variables de entrada cinemáticas y diferenciales se obtienen generalmente de los catálogos del manipulador. Las variables dinámicas deben determinarse mediante una serie de procedimientos los cuales fueron descritos en la sección (1.4.3). Por último las variables de trayectoria pueden ser obtenidas directamente de archivos propios de manipuladores, por ejemplo los archivos de extensión SRC y DAT generados por el programa comercial Robotworks para transmitir información de trayectorias a los manipuladores KUKA. Estos archivos contienen información de los datos de la herramienta, trayectoria y movimiento del manipulador y son usados para cargar la trayectoria en los controladores de estos robots. Los archivos pueden ser leídos con un editor de notas, razón por la cual es posible obtener información de la trayectoria y orientación de la herramienta.

2.7.2 VARIABLES SALIDA

Las variables de salida del algoritmo de optimización fueron reunidas en dos matrices. La primera matriz contiene la trayectoria optimizada y validada en conjunto con una serie de datos adicionales de la trayectoria que son útiles para el análisis de los resultados. La segunda matriz contiene el resumen del comportamiento del algoritmo en cada iteración a medida que se va acercando a los valores optimizados.

La matriz que almacena la trayectoria optimizada ($T_{\text{Optim}}()$) es una matriz de tamaño $((n+2) \times 64)$, siendo n el total de puntos de la trayectoria, e i el punto i-ésimo de la trayectoria. En las siguientes

tablas se ilustran como está conformada dicha matriz, según los valores obtenidos por los parámetros cinemáticos, diferenciales, y dinámicos.

PARÁMETROS CINEMÁTICOS			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Encabezados trayectoria	$T_{Optim}(1,1:64)$	----	Variable tipo texto
Punto i-ésimo de la trayectoria	$T_{Optim}(i+1,1)$	----	Variable Real Entera
Coordenada X optimizada efector final Punto i	$T_{Optim}(i+1,2)$	m	Variable Real
Coordenada Y optimizada efector final Punto i	$T_{Optim}(i+1,3)$	m	Variable Real
Coordenada Z optimizada efector final Punto i	$T_{Optim}(i+1,4)$	m	Variable Real
Angulo de Euler ϕ orientación efector final Punto i	$T_{Optim}(i+1,5)$	rad	Variable Real
Angulo de Euler γ orientación efector final Punto i	$T_{Optim}(i+1,6)$	rad	Variable Real
Angulo de Euler ψ orientación efector final Punto i	$T_{Optim}(i+1,7)$	rad	Variable Real
Angulo articulación 1 (θ_1) Punto i	$T_{Optim}(i+1,8)$	rad	Variable Real
Angulo articulación 2 (θ_2) Punto i	$T_{Optim}(i+1,9)$	rad	Variable Real
Angulo articulación 3 (θ_3) Punto i	$T_{Optim}(i+1,10)$	rad	Variable Real
Angulo articulación 4 (θ_4) Punto i	$T_{Optim}(i+1,11)$	rad	Variable Real
Angulo articulación 5 (θ_5) Punto i	$T_{Optim}(i+1,12)$	rad	Variable Real
Angulo articulación 6 (θ_6) Punto i	$T_{Optim}(i+1,13)$	rad	Variable Real
Comprobación articulación 1 Punto i	$T_{Optim}(i+1,14)$	----	Variable Real Entera
Comprobación articulación 2 Punto i	$T_{Optim}(i+1,15)$	----	Variable Real Entera
Comprobación articulación 3 Punto i	$T_{Optim}(i+1,16)$	----	Variable Real Entera
Comprobación articulación 4 Punto i	$T_{Optim}(i+1,17)$	----	Variable Real Entera
Comprobación articulación 5 Punto i	$T_{Optim}(i+1,18)$	----	Variable Real Entera
Comprobación articulación 6 Punto i	$T_{Optim}(i+1,19)$	----	Variable Real Entera
Sumatoria Comprobación articulación 1 total puntos trayectoria	$T_{Optim}(n+2,14)$	----	Variable Real Entera
Sumatoria Comprobación articulación 2 total puntos trayectoria	$T_{Optim}(n+2,15)$	----	Variable Real Entera
Sumatoria Comprobación articulación 3 total puntos trayectoria	$T_{Optim}(n+2,16)$	----	Variable Real Entera
Sumatoria Comprobación articulación 4 total puntos trayectoria	$T_{Optim}(n+2,17)$	----	Variable Real Entera
Sumatoria Comprobación articulación 5 total puntos trayectoria	$T_{Optim}(n+2,18)$	----	Variable Real Entera
Sumatoria Comprobación articulación 6 total puntos trayectoria	$T_{Optim}(n+2,19)$	----	Variable Real Entera

Tabla 8 Variables salida parámetros cinemáticos manipuladores antropomórficos de seis grados de libertad y muñeca esférica

PARÁMETROS DIFERENCIALES			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Velocidad articulación 1 ($\dot{\theta}_1$)	$T_{Optim}(i+1,20)$	rad/s	Variable Real
Velocidad articulación 2 ($\dot{\theta}_2$)	$T_{Optim}(i+1,21)$	rad/s	Variable Real
Velocidad articulación 3 ($\dot{\theta}_3$)	$T_{Optim}(i+1,22)$	rad/s	Variable Real
Velocidad articulación 4 ($\dot{\theta}_4$)	$T_{Optim}(i+1,23)$	rad/s	Variable Real
Velocidad articulación 5 ($\dot{\theta}_5$)	$T_{Optim}(i+1,24)$	rad/s	Variable Real
Velocidad articulación 6 ($\dot{\theta}_6$)	$T_{Optim}(i+1,25)$	rad/s	Variable Real
Comprobación vel. articulación 1	$T_{Optim}(i+1,26)$	----	Variable Real Entera
Comprobación vel. articulación 2	$T_{Optim}(i+1,27)$	----	Variable Real Entera
Comprobación vel. articulación 3	$T_{Optim}(i+1,28)$	----	Variable Real Entera
Comprobación vel. articulación 4	$T_{Optim}(i+1,29)$	----	Variable Real Entera
Comprobación vel. articulación 5	$T_{Optim}(i+1,30)$	----	Variable Real Entera
Comprobación vel. articulación 6	$T_{Optim}(i+1,31)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 1 total puntos trayectoria	$T_{Optim}(n+2,26)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 2 total puntos trayectoria	$T_{Optim}(n+2,27)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 3 total puntos trayectoria	$T_{Optim}(n+2,28)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 4 total puntos trayectoria	$T_{Optim}(n+2,29)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 5 total puntos trayectoria	$T_{Optim}(n+2,30)$	----	Variable Real Entera
Sumatoria Comprobación vel. articulación 6 total puntos trayectoria	$T_{Optim}(n+2,31)$	----	Variable Real Entera
Aceleración articulación 1 ($\ddot{\theta}_1$)	$T_{Optim}(i+1,32)$	rad/s ²	Variable Real
Aceleración articulación 2 ($\ddot{\theta}_2$)	$T_{Optim}(i+1,33)$	rad/s ²	Variable Real
Aceleración articulación 3 ($\ddot{\theta}_3$)	$T_{Optim}(i+1,34)$	rad/s ²	Variable Real
Aceleración articulación 4 ($\ddot{\theta}_4$)	$T_{Optim}(i+1,35)$	rad/s ²	Variable Real
Aceleración articulación 5 ($\ddot{\theta}_5$)	$T_{Optim}(i+1,36)$	rad/s ²	Variable Real
Aceleración articulación 6 ($\ddot{\theta}_6$)	$T_{Optim}(i+1,37)$	rad/s ²	Variable Real

Tabla 9 Variables salida parámetros diferenciales manipuladores antropomórficos de seis grados de libertad y muñeca esférica

PARÁMETROS DINÁMICOS			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Torque articulación 1 (τ_1)	$T_{Optim}(i+1,38)$	Nm	Variable Real
Torque articulación 2 (τ_2)	$T_{Optim}(i+1,39)$	Nm	Variable Real
Torque articulación 3 (τ_3)	$T_{Optim}(i+1,40)$	Nm	Variable Real
Torque articulación 4 (τ_4)	$T_{Optim}(i+1,41)$	Nm	Variable Real
Torque articulación 5 (τ_5)	$T_{Optim}(i+1,42)$	Nm	Variable Real
Torque articulación 6 (τ_6)	$T_{Optim}(i+1,43)$	Nm	Variable Real
Torque motor 1 (τ_{m1})	$T_{Optim}(i+1,44)$	Nm	Variable Real
Torque motor 2 (τ_{m2})	$T_{Optim}(i+1,45)$	Nm	Variable Real
Torque motor 3 (τ_{m3})	$T_{Optim}(i+1,46)$	Nm	Variable Real
Torque motor 4 (τ_{m4})	$T_{Optim}(i+1,47)$	Nm	Variable Real
Torque motor 5 (τ_{m5})	$T_{Optim}(i+1,48)$	Nm	Variable Real
Torque motor 6 (τ_{m6})	$T_{Optim}(i+1,49)$	Nm	Variable Real
Potencia motor 1 (P_1)	$T_{Optim}(i+1,50)$	watt	Variable Real
Potencia motor 2 (P_2)	$T_{Optim}(i+1,51)$	watt	Variable Real
Potencia motor 3 (P_3)	$T_{Optim}(i+1,52)$	watt	Variable Real
Potencia motor 4 (P_4)	$T_{Optim}(i+1,53)$	watt	Variable Real
Potencia motor 5 (P_5)	$T_{Optim}(i+1,54)$	watt	Variable Real
Potencia motor 6 (P_6)	$T_{Optim}(i+1,55)$	Kw	Variable Real
Potencia total punto i (Pot)	$T_{Optim}(i+1,56)$	Kw	Variable Real
Comprobación Potencia Total punto i	$T_{Optim}(i+1,57)$	----	Variable Real Entera
Sumatoria Comprobación Potencia	$T_{Optim}(n+2,57)$	----	Variable Real Entera

Tabla 10 Variables salida parámetros dinámicos manipuladores antropomórficos de seis grados de libertad y muñeca esférica

VARIABLES DE SALIDA OPTIMIZADAS			
VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Manipulabilidad brazo (w_A)	$T_{Optim}(i+1,58)$	----	Variable Real
Manipulabilidad muñeca (w_W)	$T_{Optim}(i+1,59)$	----	Variable Real
$1 - w$	$T_{Optim}(i+1,60)$	----	Variable Real
$1 - w$ promedio de trayectoria	$T_{Optim}(n+2,60)$	----	Variable Real
Magnitud punto i-1 a punto i ($ u $)	$T_{Optim}(i+1,61)$	m	Variable Real
Velocidad lineal punto i-1 a punto i (VT)	$T_{Optim}(i+1,62)$	m/s	Variable Real
Tiempo punto i-1 a punto i (t)	$T_{Optim}(i+1,63)$	s	Variable Real

Tiempo total trayectoria (t_T)	$T_{Optim}(n+2,63)$	s	Variable Real
Consumo punto i-1 a punto i (C_i)	$T_{Optim}(i+2,64)$	KWH	Variable Real
Consumo total trayectoria (C)	$T_{Optim}(n+2,64)$	KWH	Variable Real

Tabla 11 Variables salida optimizadas manipuladores antropomórficos de seis grados de libertad y muñeca esférica

La matriz que almacena el comportamiento del algoritmo en cada iteración (HKA()) es una matriz de tamaño (8xn), siendo n tres veces el total de las k iteraciones más uno, necesarias para optimizar la trayectoria. El valor i equivale a la iteración i-ésima del algoritmo. En la siguiente tabla se detallan las salidas de dicha matriz.

VARIABLE	IDENTIFICADOR	UNIDADES	TIPO
Encabezados matriz HKA	HKA(1,i*3)	----	Variable tipo texto
Iteración i	HKA(2,i*3)	----	Variable Real Entera
Radio del espacio de trabajo iteración i (R_i)	HKA(3,i*3)	m	Variable Real
Angulo de colatitud espacio de trabajo iteración i (γ_i)	HKA(3,i*3+1)	rad	Variable Real
Angulo azimutal espacio de trabajo iteración i (ϕ_i)	HKA(3,i*3+2)	rad	Variable Real
σ_1 de matriz de varianzas \sum_i iteración i	HKA(4,i*3)	m	Variable Real
σ_2 de matriz de varianzas \sum_i iteración i	HKA(4,i*3+1)	m	Variable Real
σ_3 de matriz de varianzas \sum_i iteración i	HKA(4,i*3+2)	m	Variable Real
Función de Costo (Jq) Iteración i	HKA(5,i*3)	----	Variable Real
Consumo total trayectoria iteración i	HKA(6,i*3)	KWH	Variable Real
Manipulabilidad total trayectoria iteración i	HKA(7,i*3)	----	Variable Real
Tiempo total trayectoria iteración i	HKA(8,i*3)	s	Variable Real

Tabla 12 Salidas matriz que refleja comportamiento del algoritmo de optimización en k iteraciones para manipuladores antropomórficos de seis grados de libertad y muñeca esférica.

2.7.3 DIAGRAMA FLUJO SIMPLIFICADO ALGORITMO PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD

A continuación se expone el diagrama de flujo simplificado del algoritmo de planeación de trayectorias. Inicialmente se presenta un diagrama de flujo inicial (Fig. 21) que contiene la forma como están enlazadas las funciones principales desarrolladas. Estas son la función Espacio_Trab, la cual permite calcular el espacio de trabajo del robot una vez este se ha seleccionado, la función Valid_Tray, la cual permite validar la trayectoria seleccionada calculando sus parámetros de

manipulabilidad y consumo eléctrico (al igual que otros parámetros de apoyo como desplazamientos, velocidades, torques, entre otros), y la función Opt_HKA, la cual se encarga de calcular la trayectoria optimizada según los datos de entrada generados por las otras dos funciones.

El esquema de esta función se explica en detalle en las Fig. 22, Fig. 23, y Fig. 24.

Con estos diagramas de flujo se pretende dar a conocer la forma cómo interactúan entre si cada uno de los temas tratados en el proyecto convergiendo en un método final de planeación de trayectorias. . En el Anexo I está disponible el algoritmo en pseudocódigo el cual permite profundizar en el funcionamiento del algoritmo al igual que de las funciones que lo conforman.

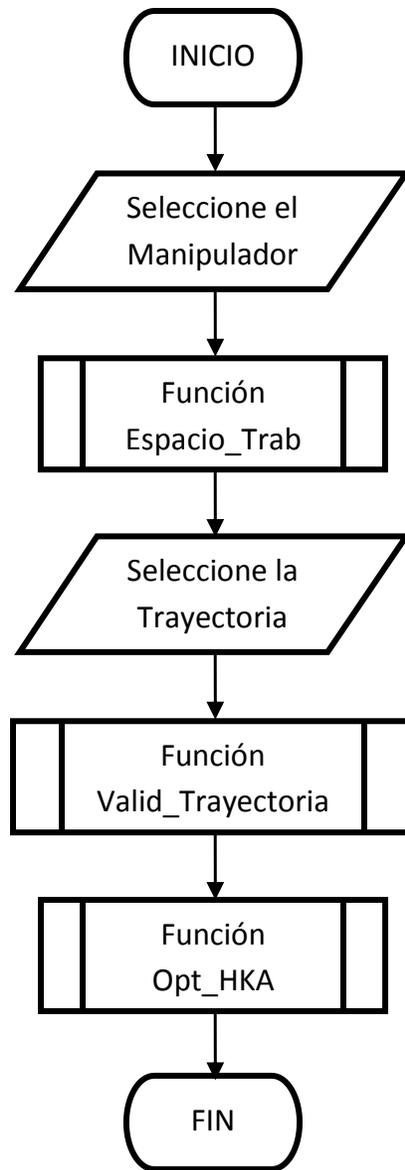


Figura 21 Diagrama de flujo funciones principales algoritmo de planeación de trayectorias

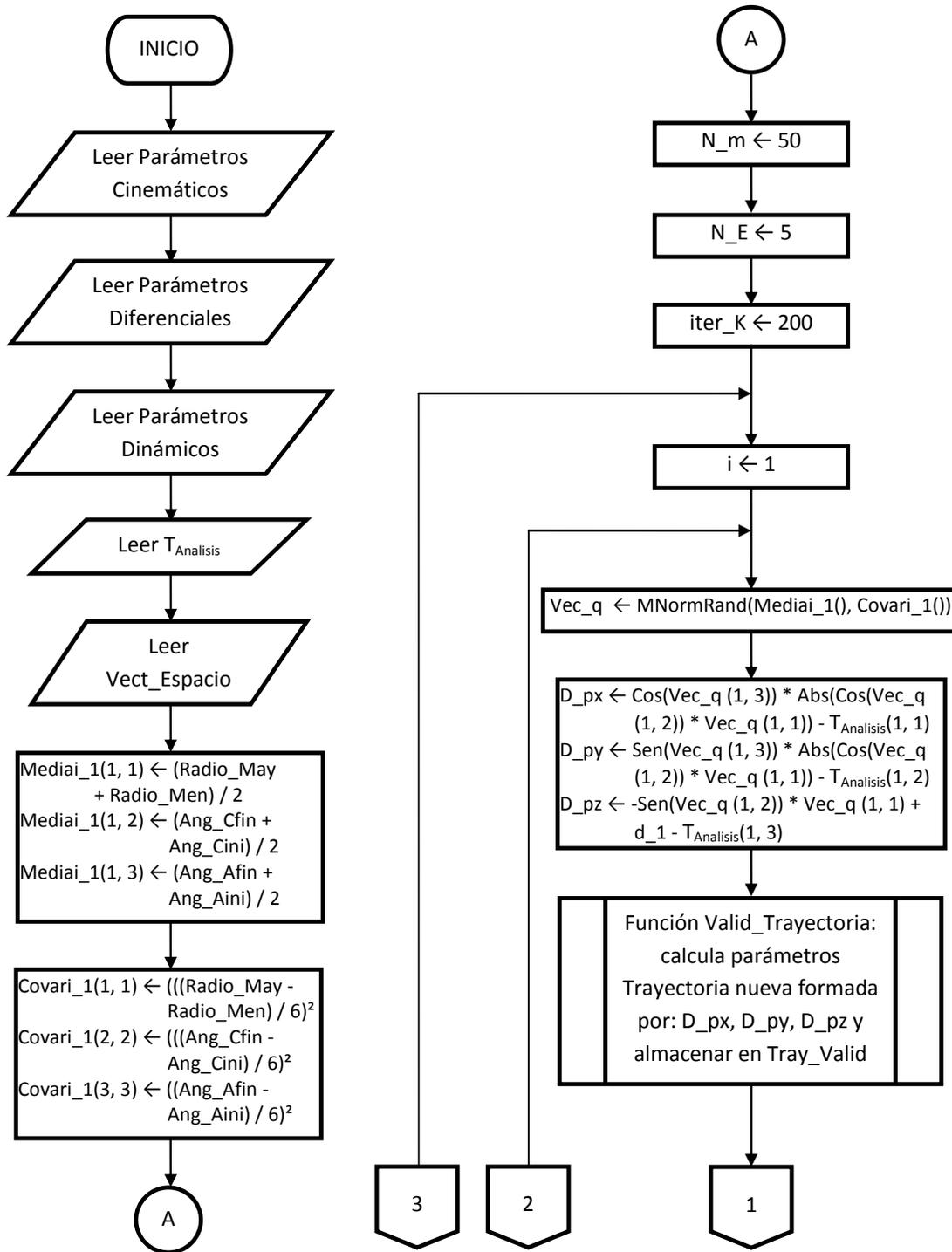


Figura 22 Diagrama de flujo función Opt_HKA (parte 1)

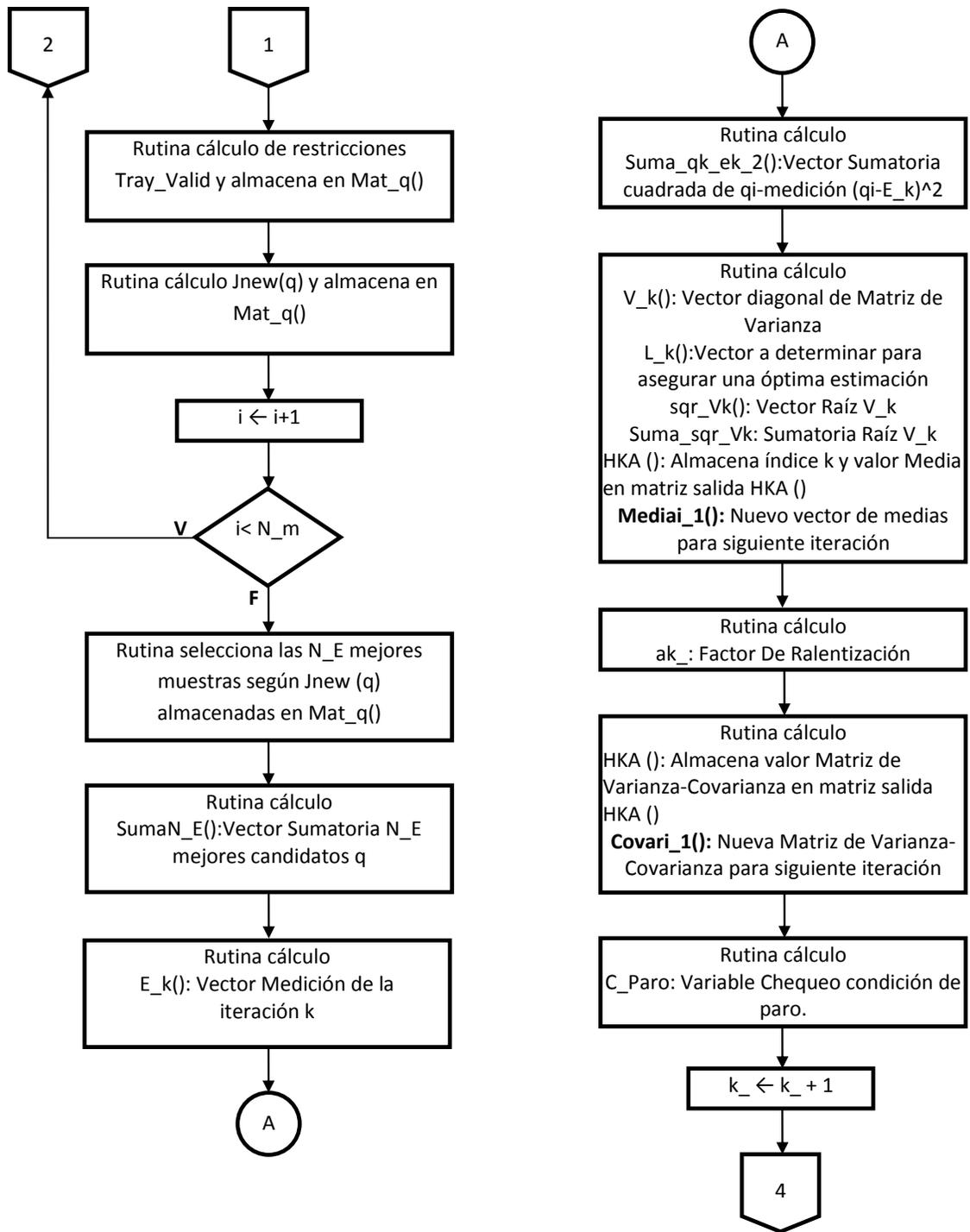


Figura 23 Diagrama de flujo función Opt_HKA (parte 2)

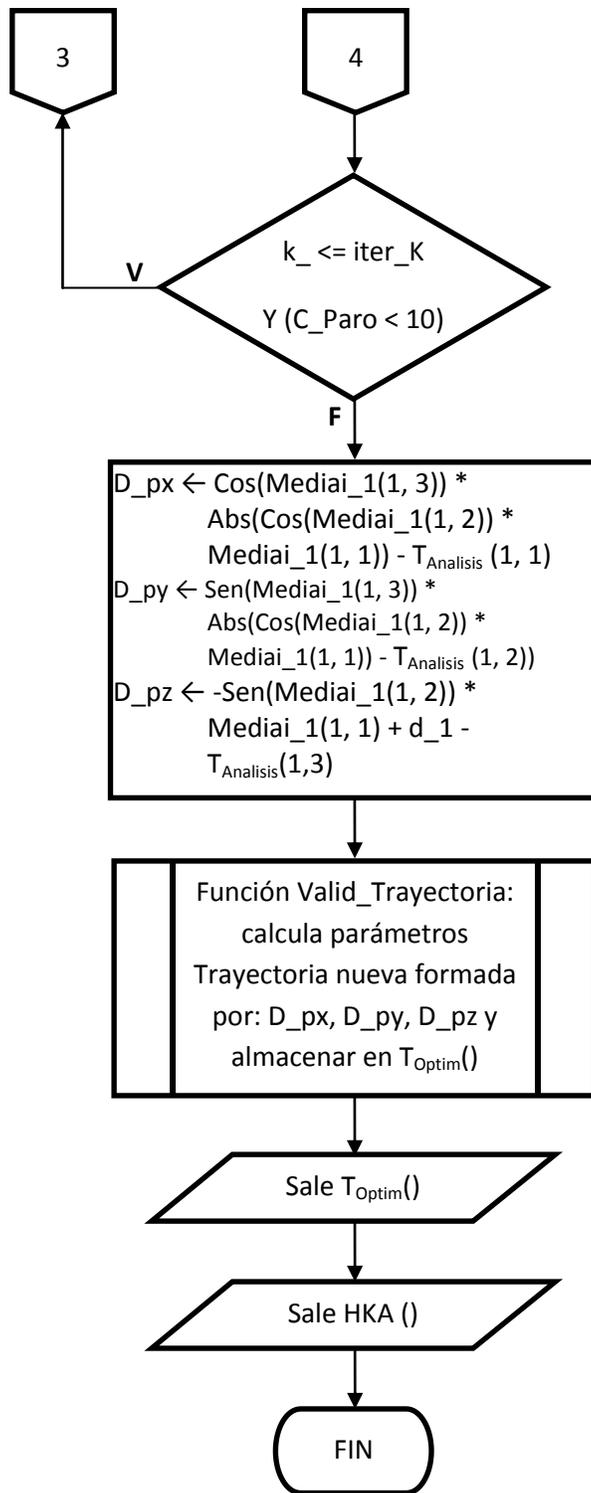


Figura 24 Diagrama de flujo función Opt_HKA (parte 3)

2.7.4 DIAGRAMA DE FLUJO SIMPLIFICADO FUNCIÓN VALIDACIÓN TRAYECTORIA

La función encargada de validar cuando una trayectoria puede ser ejecutada por el manipulador se llama Valid_Tray. Esta función transforma en coordenadas articulares la trayectoria que se desea analizar, posteriormente calcula la cinemática directa, la cinemática diferencial y dinámica inversa, chequeando que tanto las coordenadas articulares, velocidades de articulación y potencia total se encuentren dentro de los límites propios del robot.

El diagrama de flujo de esta función se expone en las Fig. 25 y Fig. 26.

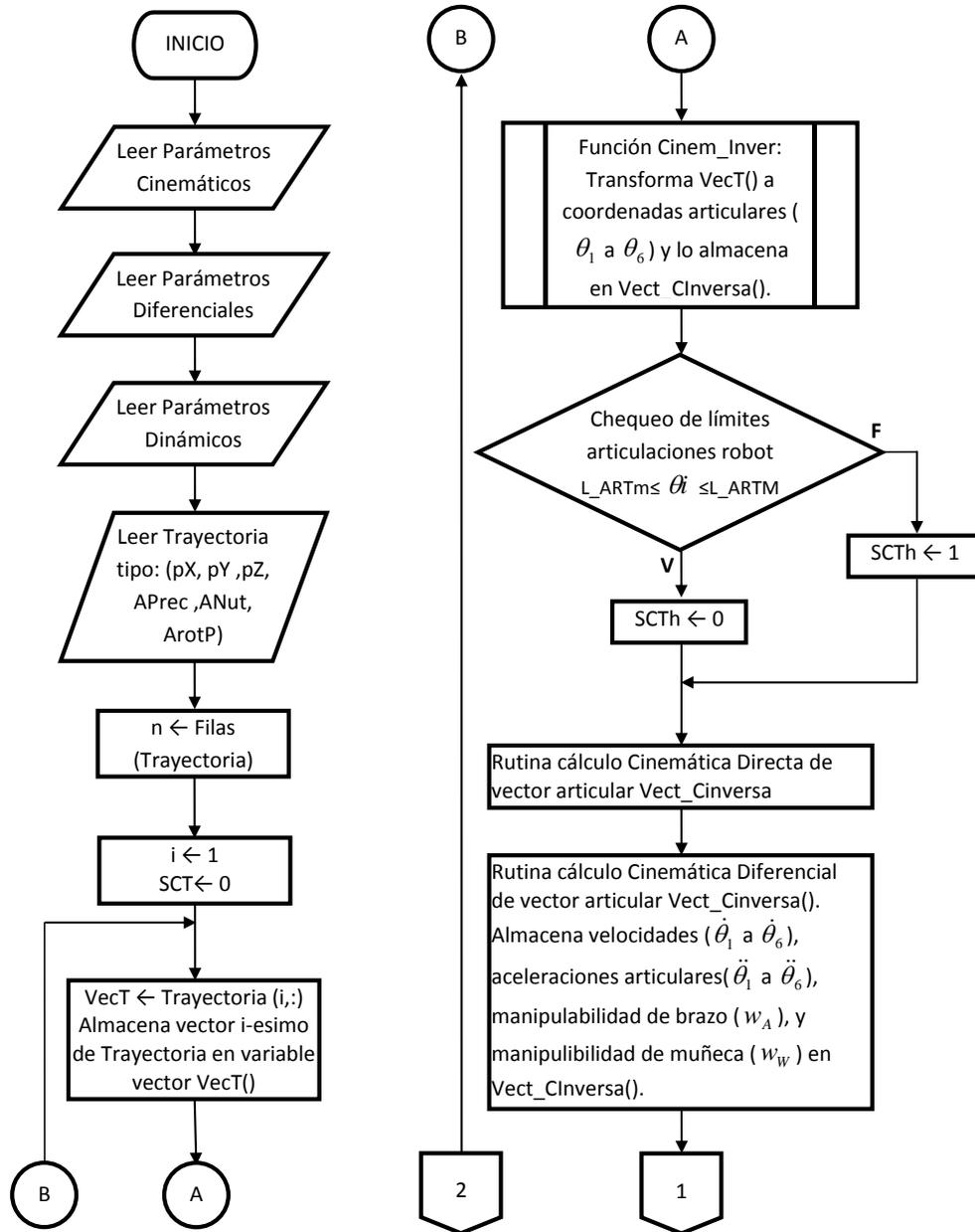


Figura 25 Diagrama de flujo función para validación trayectoria (parte 1)

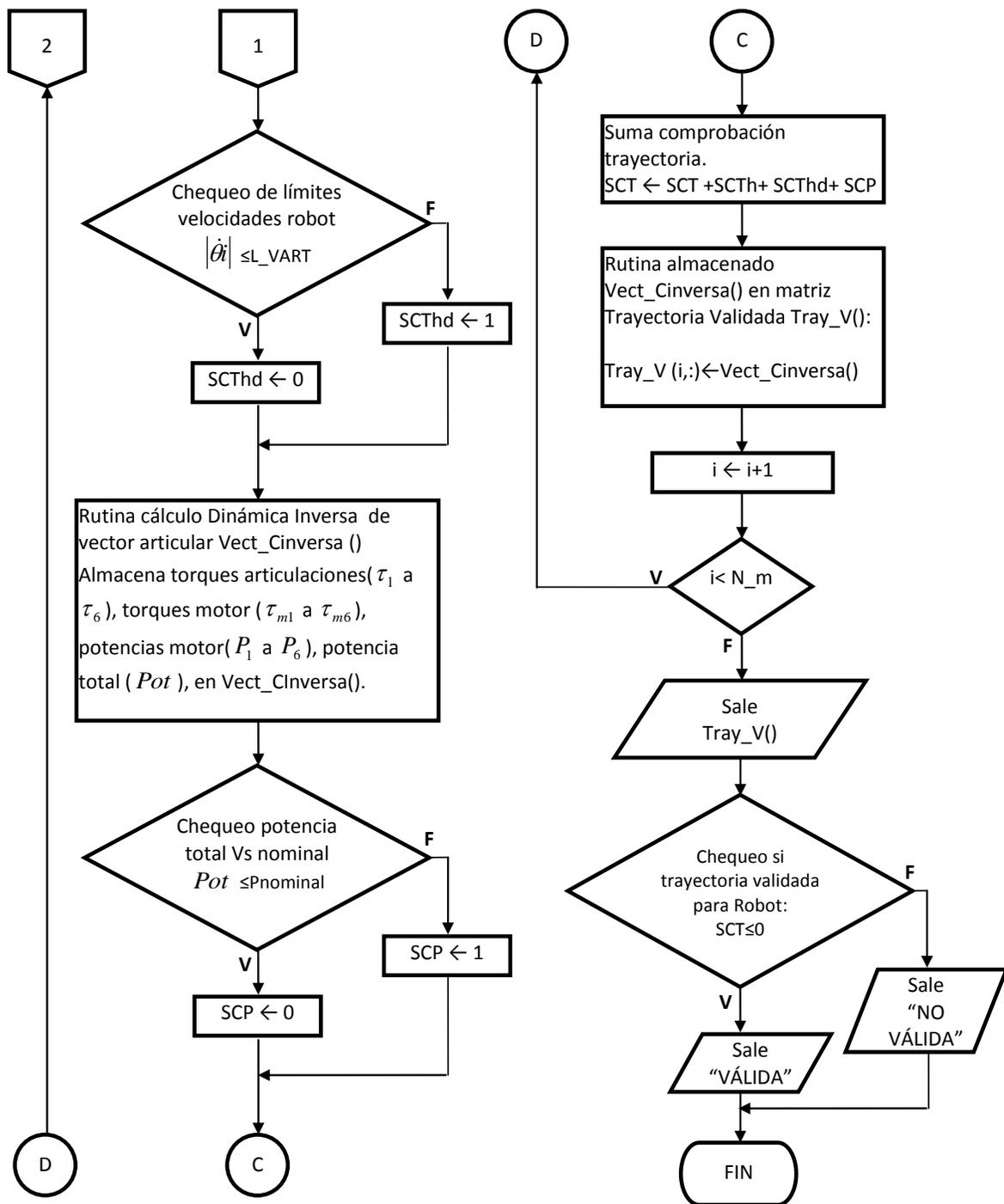


Figura 26 Diagrama de flujo función para validación trayectoria (parte 2)

2.7.5 PROGRAMA DE PLANEACIÓN DE TRAYECTORIAS EN MANIPULADORES SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA BASADA EN OPTIMIZACIÓN DE ENERGÍA ELÉCTRICA Y MANIPULABILIDAD

El programa final fue implementado en la aplicación de Microsoft Visual Basic que posee el software CAD Autodesk Inventor® 2013.

Las entradas del programa son archivos en Excel clasificados de la siguiente forma:

Trayectorias: Archivos con las variables de la trayectoria a Optimizar

Parámetros: Archivos con los parámetros de cinemática directa, diferencial, y dinámica del manipulador.

Igualmente las salidas del programa son archivos en Excel clasificados de la siguiente forma:

Trayectorias con Parámetros: Archivos con las variables de la trayectoria a Optimizar y sus parámetros sin optimizar.

Trayectorias Optimizadas: Archivos con las variables de la trayectoria optimizada y parámetros optimizados.

HKA: archivos con la información de las función de costo, consumo, manipulabilidad y ubicación en el espacio de trabajo simplificado de cada iteración realizada necesaria para alcanzar una solución óptima.

El programa consta de un control multi-páginas, compuesto por 5 páginas, las cuales contienen:

2.7.5.1 Página 1: INICIO

En esta página se encuentra un control desplegable principal (ComboBox) el cual permite al usuario seleccionar entre los robots disponibles.

Adicionalmente cuenta con dos controles: uno para cargar los parámetros del robot seleccionado, y otro para finalizar el programa.

Un pantallazo de esta página se puede apreciar en la Fig. 27.

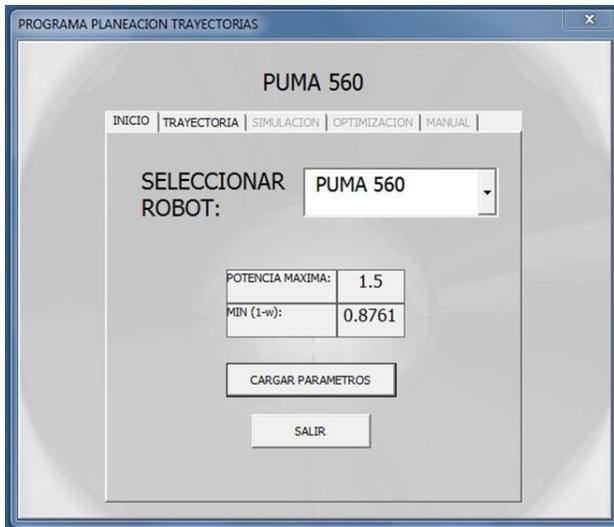


Figura 27 Página 1 programa de planeación de trayectorias

2.7.5.2 Página 2: TRAYECTORIA

En esta página se encuentra un control desplegable principal (ComboBox) el cual permite al usuario seleccionar la trayectoria con la cual desea trabajar.

Adicionalmente cuenta con dos controles: uno que se encarga de cargar la trayectoria seleccionada y calcular los criterios de ésta, y otro para guardar, cuando se requiera, la trayectoria con todos los criterios calculados.

Un pantallazo de esta página se puede apreciar en la Fig. 28.

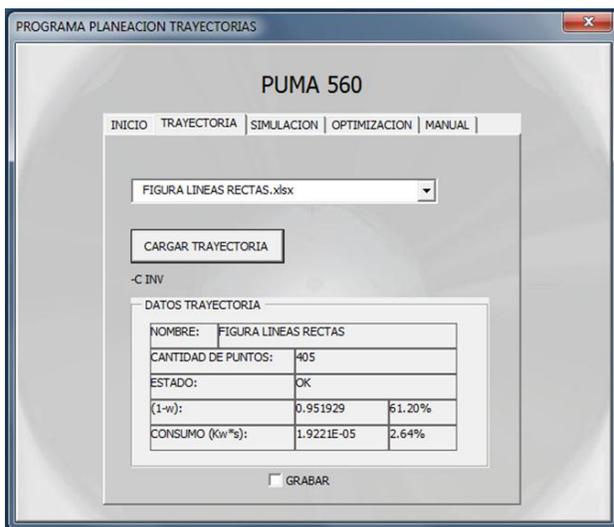


Figura 28 Página 2 programa de planeación de trayectorias

2.7.5.3 Página 3: SIMULACIÓN

Esta página cuenta con un botón de control que permite cargar en el software Autodesk Inventor el manipulador seleccionado con la figura correspondiente a la trayectoria a realizar.

Posee también dos botones de conmutación que permiten fijar si el avance es positivo o negativo en el efector final.

Y por último posee un botón tipo flecha que permite escoger entre tres velocidades (alta, media y baja) de ejecución de la trayectoria.

Un pantallazo de esta página se puede apreciar en la Fig. 29.

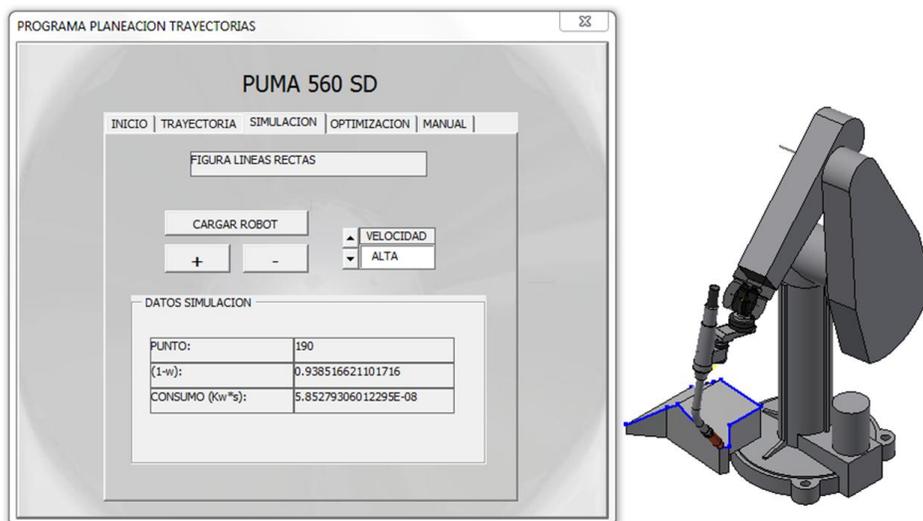


Figura 29 Página 3 programa de planeación de trayectorias

2.7.5.4 Página 4: OPTIMIZACIÓN

Esta página cuenta con un cajón de texto que permite introducir por teclado el peso deseado para el criterio de consumo eléctrico, y también posee un control que permite la ejecución de la rutina de optimización de trayectorias.

Un pantallazo de esta página se puede apreciar en la Fig. 30.

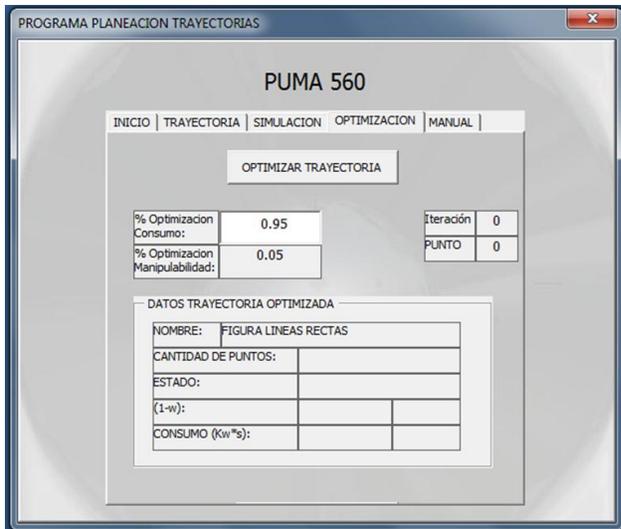


Figura 30 Página 4 programa de planeación de trayectorias

2.7.5.5 Página 5: MANUAL

En esta página se pueden calcular manualmente dentro del espacio de trabajo simplificado trayectorias derivadas de la cargada, variando los valores de los tres cajones de texto correspondientes al radio, ángulo de colatitud y ángulo azimutal del espacio de trabajo.

Adicionalmente cuenta con dos controles: uno que se encarga de calcular los criterios de la trayectoria con los valores de espacio de trabajo indicados, y otro para guardar, cuando se requiera, la trayectoria encontrada con todos los criterios calculados.

Un pantallazo de esta página se puede apreciar en la Fig. 31.

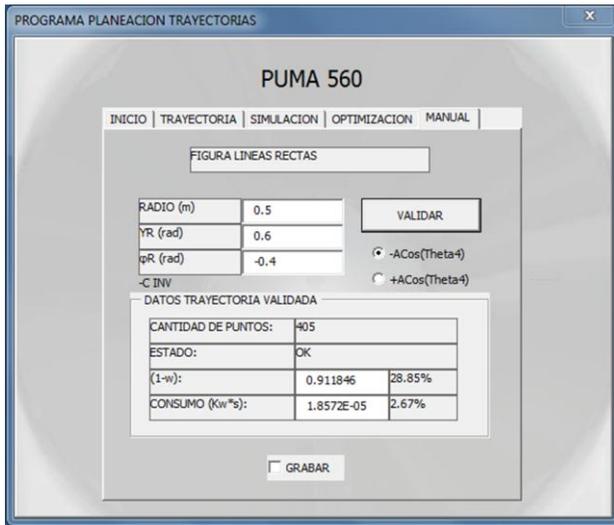


Figura 31 Página 4 programa de planeación de trayectorias

3 RESULTADOS

En este capítulo se hace una presentación de los resultados obtenidos a partir de simulaciones y optimización de trayectorias de manipuladores de muñeca esférica de seis grados de libertad, implementadas en una aplicación de Visual Basic perteneciente al Software Autodesk Inventor 2013® el cual es un software CAD usado en la industria para el diseño de piezas mecánicas.

La obtención de resultados del presente proyecto partió de la planeación y optimización según criterios de manipulabilidad y consumo eléctrico de cinco trayectorias mediante el algoritmo de optimización de trayectorias desarrollado, evaluadas en dos manipuladores antropomorfos seriales de muñeca esférica de seis grados de libertad: El robot PUMA 560, y el robot KUKA KR5 HW ARC.

Para el Robot PUMA 560 se disponen de todos los parámetros dinámicos necesarios para correr el algoritmo, gracias a que estos fueron recopilados de diversas investigaciones (Armstrong, Khatib, & Burdick, 1986), (Corke P. , 1996). Debido a lo anterior fue posible realizar un paralelo del comportamiento del algoritmo en la planeación de trayectorias con manipuladores con todos los parámetros dinámicos y trayectorias sin parámetros dinámicos.

Al no contar con todos los parámetros dinámicos del robot KUKA KR5 HW ARC, fue necesario evaluar sus trayectorias con los datos usados en la configuración del algoritmo en el robot PUMA 560 sin parámetros dinámicos.

Las trayectorias usadas en la evaluación del comportamiento del algoritmo fueron generadas directamente en el espacio cartesiano, con velocidades del efector final constante (0.5m/s), salvo en algunos puntos en los cuales se debe rebajar la velocidad para no exceder los límites de las articulaciones del manipulador.

Los datos de las trayectorias optimizadas se obtuvieron mediante la aplicación del algoritmo de optimización desarrollado, basado en el algoritmo heurístico de Kalman, el cual se enfoca en la minimización del criterio de consumo eléctrico y maximización de la manipulabilidad.

Tomando como punto de partida las trayectorias Robotworks y la trayectoria Tarro Inclinado, se identificaron los parámetros necesarios para que el algoritmo arroje optimizaciones aceptables. Estos parámetros son:

Número máximo de iteraciones: se estableció experimentalmente en 200 debido a que ninguna de las trayectorias optimizadas tomó más de esta cantidad para alcanzar un valor óptimo en sus parámetros según la regla de paro establecida.

El peso asignado para cada criterio de optimización involucrados en la función de costo a minimizar, fue establecido en 0.95 para el consumo de energía y 0.05 para la manipulabilidad en robots con todos los parámetros dinámicos conocidos, y de 0.9995 para consumo de energía y 0.0005 para la manipulabilidad en robots para los cuales solo se conocen sus parámetros cinemáticos.

El valor idóneo del coeficiente de ralentización del algoritmo: se estableció experimentalmente en 0.9, debido a que coeficientes menores aumentaban la cantidad de iteraciones necesarias para obtener valores óptimos establecidos en la regla de paro del algoritmo.

Para que se dé la regla de paro del algoritmo en 10 iteraciones seguidas se debe cumplir con las siguientes condiciones:

El valor absoluto de la diferencia entre la manipulabilidad minimizada de la iteración actual y la de la iteración anterior debe ser menor a 0.0005.

La función de costo de la iteración actual debe ser menor a 0.9

El valor absoluto de la diferencia entre el consumo eléctrico de la iteración actual y la de la iteración anterior debe ser menor a 0.0000001.

El valor absoluto de la diferencia entre el porcentaje de consumo eléctrico de la iteración actual y la de la iteración anterior debe ser menor a 0.0005.

A continuación se hace una descripción del manipulador PUMA 560 y KUKA KR5 HW ARC al igual que de cada una de las trayectorias, y se exponen los resultados obtenidos partiendo de la aplicación del algoritmo de planeación.

3.1 MANIPULADORES USADOS CON ALGORITMO DE PLANEACIÓN DE TRAYECTORIAS

Los manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica seleccionados para validar el algoritmo fueron el manipulador PUMA 560 y el manipulador KUKA KR5 KW ARC. A continuación se hace una descripción de cada uno de estos.

3.1.1 Manipulador UNIMATE PUMA 560

El manipulador PUMA (Programmable Universal Machine for Assembly) es un robot fabricado en 1978 por Unimation, basado en un estudio realizado para General Motors (Spong, Hutchinson, & Vidyasagar, 2004). Los manipuladores de la serie UNIMATE PUMA 560 se caracterizan por ser compactos, característica que los hace ideales para la ejecución de tareas de liviano a pesado rendimiento tales como ensamble, soldadura, manejo de materiales, empaque y aplicaciones de inspección. Estos son manipuladores de seis grados de libertad con todas sus articulaciones rotacionales. También se caracterizan por tener su muñeca esférica debido a que los ejes de sus tres últimas articulaciones se intersecan en un punto común (Craig, 2005).

El PUMA 560 cuenta en su eje final (articulación 6) con un acople que permite adaptar la herramienta necesaria para la tarea a realizar. Para esta investigación se escoge como herramienta final la pistola de soldadura de arco Tough Gun G2 del fabricante Treggaskiss la cual cuenta con el acople correspondiente y peso ideal para adaptarla al manipulador PUMA 560 (UNIMATION, 1984), (Treggaskiss, 2012).

El modelo cinemático de este manipulador se basa en la convención DH para manipuladores antropomórficos de seis grados de libertad y muñeca esférica. Partiendo de las dimensiones de las articulaciones para este manipulador y la pistola, tomadas directamente del catálogo de los fabricantes (UNIMATION, 1984), (Tregaskiss, 2012), e identificando las variables de la convención DH en un esquema del PUMA 560 (Fig. 32) se procede a plantear su modelo cinemático, ilustrado en la Tabla 13

Articulaciones (θ_i)	d_i (m)	a_i (m)	α_i ($^\circ$)
θ_1	0.67183	0	-90
θ_2	0	0.4318	0
θ_3	0.1397	0	-90
θ_4	0.4318	0	90
θ_5	0	0	-90
θ_6	0.05588 + 0.4	0	0

Tabla 13 Modelo cinemático basado en la convención DH para manipulador PUMA 560

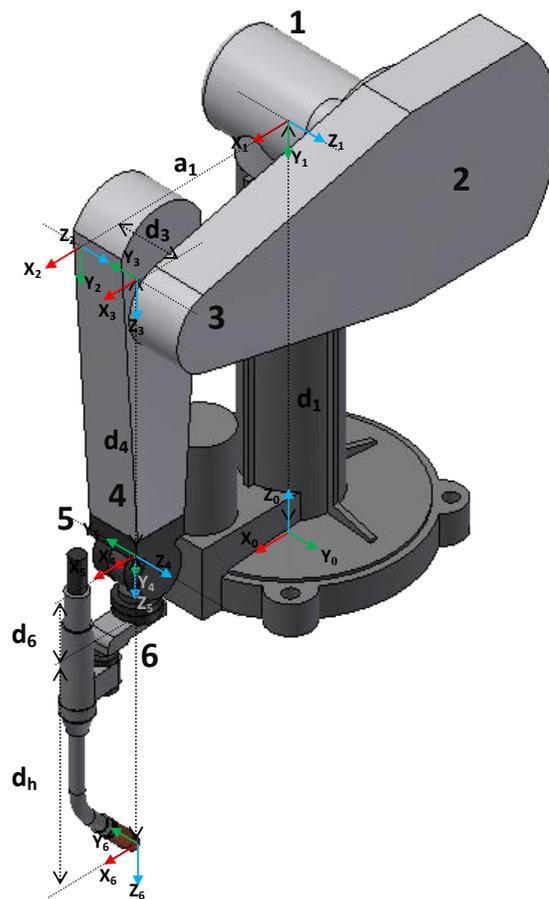


Figura 32 Esquema convención DH PUMA 560

Los límites de las articulaciones y el límite de velocidad máxima del efector final también se obtienen del catálogo del fabricante.

Articulación	LIMITE INF (°)	LIMITE SUP (°)
1	-160	160
2	-223	43
3	-232	52
4	-110	170
5	-100	100
6	-150	150

Tabla 14 Límite articulaciones PUMA 560

Velocidad Lineal:	0.5 m/seg
-------------------	-----------

Tabla 15 Velocidad lineal máxima efector final PUMA 560

Los límites de las velocidades articulares y los parámetros dinámicos del manipulador PUMA 560 fueron extraídos de diversas investigaciones que se especializaron en la identificación de cada uno de estos parámetros (Armstrong, Khatib, & Burdick, 1986), (Corke P. , 1996), (Corke P. I., 1996). A continuación se presentan en tablas los valores usados en la ejecución del algoritmo:

Articulación	LIMITE (°/s)
1	120
2	163
3	129
4	406
5	366
6	440

Tabla 16 Límite velocidades articulares PUMA 560

Eslabones	SX (m)	SY (m)	SZ (m)
1	0	0	0
2	0.068	0.006	-0.016
3	0	-0.07	0.014
4	0	0	-0.019
5	0	0	0
6	0	0	0.032

Tabla 17 Centro de masa eslabones PUMA 560

Eslabones	mi (Kg)
-----------	---------

1	31.04
2	17.4
3	4.8
4	0.82
5	0.35
6	0.09

Tabla 18 Peso eslabones PUMA 560

Momentos de Inercia Eslabón 1 (Kg.m ²)		
0	0	0
0	0	0
0	0	0
Momentos de Inercia Eslabón 2 (Kg.m ²)		
0.13	0	0
0	0.524	0
0	0	0.539
Momentos de Inercia Eslabón 3 (Kg.m ²)		
0.066	0	0
0	0.0125	0
0	0	0.086
Momentos de Inercia Eslabón 4 (Kg.m ²)		
1.80E-03	0	0
0	1.80E-03	0
0	0	1.30E-03
Momentos de Inercia Eslabón 5 (Kg.m ²)		
3.00E-04	0	0
0	3.00E-04	0
0	0	4.00E-04
Momentos de Inercia Eslabón 6 (Kg.m ²)		
1.50E-04	0	0
0	1.50E-04	0
0	0	4.00E-05

Tabla 19 Momentos de inercia eslabones PUMA 560

Motor 1 (N.m.s/rad)	Motor 2 (N.m.s/rad)	Motor 3 (N.m.s/rad)	Motor 4 (N.m.s/rad)	Motor 5 (N.m.s/rad)	Motor 6 (N.m.s/rad)
5.8	37	6	0.1	1.9	0.7

Tabla 20 Coeficientes Viscosidad Motores Robot PUMA 560

Motor 1 (N.m/A)	Motor 2 (N.m/A)	Motor 3 (N.m/A)	Motor 4 (N.m/A)	Motor 5 (N.m/A)	Motor 6 (N.m/A)
0.223	0.226	0.24	0.069	0.072	0.066

Tabla 21 Constante Torque Motores Robot PUMA 560

Motor 1 (V.s/rad)	Motor 2 (V.s/rad)	Motor 3 (V.s/rad)	Motor 4 (V.s/rad)	Motor 5 (V.s/rad)	Motor 6 (V.s/rad)
0.223	0.226	0.24	0.069	0.072	0.066

Tabla 22 Constante De Fuerza Contra Electromotriz Motores Robot PUMA 560

Motor 1 (Ohms)	Motor 2 (Ohms)	Motor 3 (Ohms)	Motor 4 (Ohms)	Motor 5 (Ohms)	Motor 6 (Ohms)
2.1	2.1	2.1	6.7	6.7	6.7

Tabla 23 Resistencia Armaduras Motores Robot PUMA 560

Motor 1	Motor 2	Motor 3	Motor 4	Motor 5	Motor 6
62.61	107.36	53.59	76.01	71.91	76.73

Tabla 24 Relación De Transmisión Motores Robot PUMA 560

La potencia nominal del manipulador se obtiene directamente del catálogo del fabricante.

Potencia Nominal:	1.5 Kw
-------------------	--------

Tabla 25 Potencia nominal PUMA 560

Los parámetros de la pistola TOUGH GUN G2(Fig. 33) se obtienen del catálogo del fabricante y estos se enuncian en las siguientes tablas:

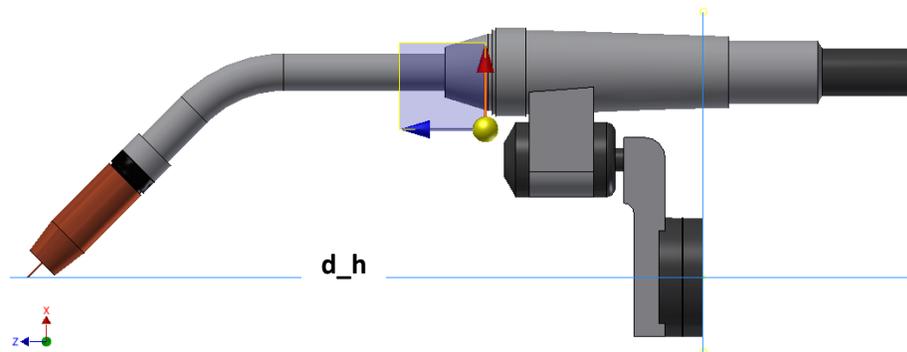


Figura 33 Pistola soldadura de arco TOUGH GUN G2

Marca:	TOUGH GUN G2 -TREGASKISS	
Distancia Horizontal (d_h):	0.4	m
Distancia Vertical (a_h):	0	m
Peso Pistola (Kg):	2.37	Kg

Tabla 26 Datos generales pistola TOUGH GUN G2

X (m)	Y (m)	Z (m)
0.08687	-7.8728E-18	0.12852

Tabla 27 Centro de masa pistola TOUGH GUN G2

Matriz Momento Inercia (Kg.m ²)		
0.032	0	-4.00E-04
0	0.037	0
-4.00E-04	0	0.005

Tabla 28 Matriz momento de inercia pistola TOUGH GUN G2

3.1.2 Manipulador KUKA KR5 HW ARC

El manipulador KUKA KR5 HW ARC es un robot fabricado por KUKA Robot GROUP. Este robot fue diseñado especialmente para realizar tareas de soldadura por arco, por lo cual presenta características propias para este proceso como una abertura de 50mm en la sección de la muñeca usada para contener las mangueras de la pistola de soldadura de arco evitando posibles interferencias con los movimientos del manipulador. Estos son manipuladores de seis grados de libertad con todas sus articulaciones rotacionales y adicionalmente su muñeca es esférica.

Para poder sujetar la herramienta (pistola) el robot KUKA KR5 HW ARC cuenta con una brida de montaje en el final de la última articulación de la muñeca. Para esta investigación se escoge como herramienta final la pistola de soldadura de arco ABIROB A del fabricante ABICOR-BINZEL la cual cuenta con la brida correspondiente y peso ideal para adaptarla al manipulador KUKA KR5 ARC HW.(ABICOR BINZEL, 2013)

El modelo cinemático de este manipulador se basa en la convención DH para manipuladores antropomórficos de seis grados de libertad y muñeca esférica. Partiendo de las dimensiones de las articulaciones para este manipulador y la pistola, tomadas directamente del catálogo de los fabricantes (Group, 2009),(ABICOR BINZEL, 2013), e identificando las variables de la convención DH en un esquema del KUKA KR5 ARC HW (Fig. 34) se procede a plantear su modelo cinemático, ilustrado en la Tabla 29.

Articulaciones (θ_i)	d_i (m)	a_i (m)	α_i ($^\circ$)
θ_1	0.4	0.18	-90
θ_2	0	0.6	0
θ_3	0	0.17	-90
θ_4	0.62	0	90
θ_5	0	0	-90
θ_6	$0.2 + 0.341$	0.107	0

Tabla 29 Modelo cinemático basado en la convención DH para manipulador KUKA KR5 ARC HW

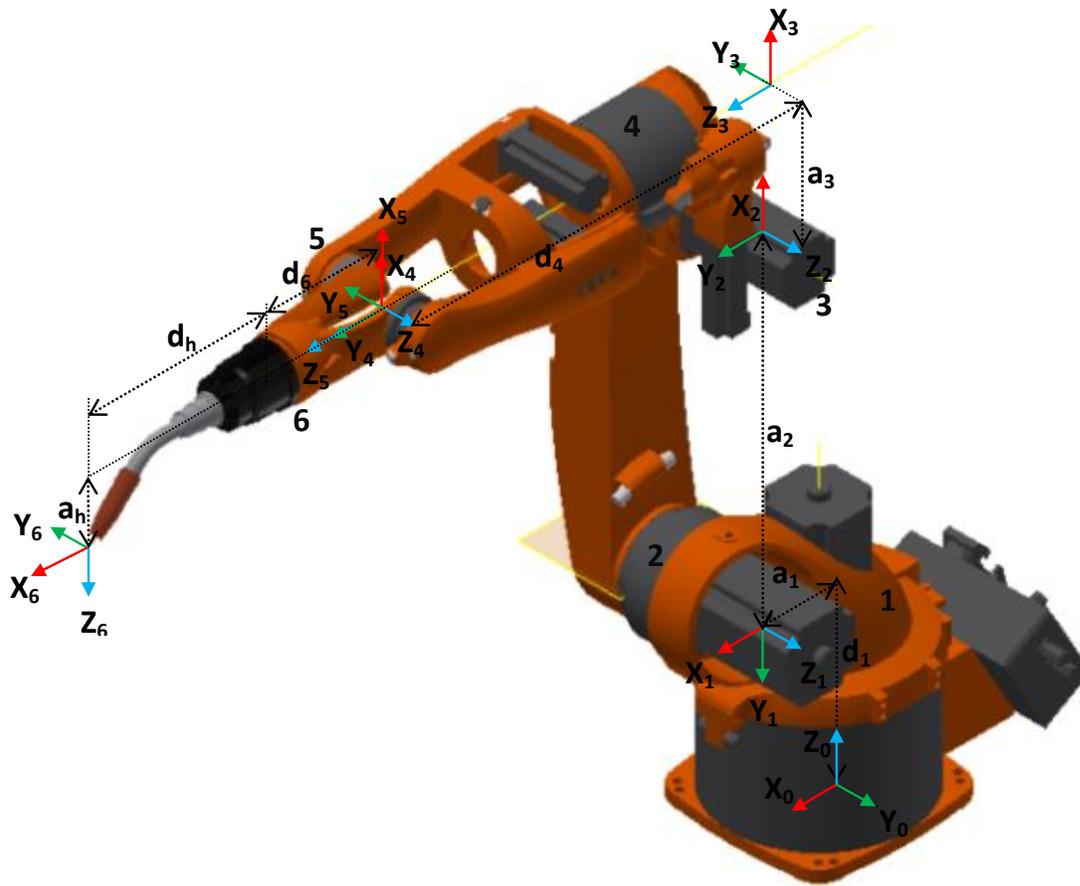


Figura 34 Esquema convención DH KUKA KR5 ARC HW

Los límites de las articulaciones, el límite de velocidad máxima del efector final, los límites de las velocidades articulares y la potencia nominal del manipulador, también se obtienen del catálogo del fabricante.

Articulación	LIMITE INF (°)	LIMITE SUP (°)
1	-160	160
2	-223	43
3	-232	52
4	-110	170
5	-100	100
6	-150	150

Tabla 30 Límite articulaciones KUKA KR5 ARC HW

Velocidad Lineal:	0.5 m/seg
-------------------	-----------

Tabla 31 Velocidad lineal máxima efector final KUKA KR5 ARC HW

Articulación	LIMITE (°/s)
1	156
2	156
3	227
4	390
5	390
6	858

Tabla 32 Límite velocidades articulares KUKA KR5 ARC HW

Potencia Nominal:	7.3 Kw
-------------------	--------

Tabla 33 Potencia nominal KUKA KR5 ARC HW

Los parámetros de la pistola ABIROB A (Fig. 35) se obtienen del catálogo del fabricante y estos se enuncian en las siguientes tablas:

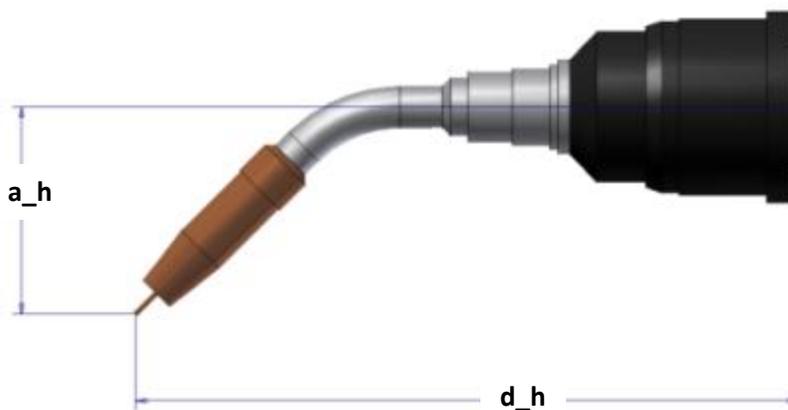


Figura 35 Pistola soldadura de arco ABIROB A

Marca:	TOUGH GUN G2 -TREGASKISS	
Distancia Horizontal (d_h):	0.340773	m
Distancia Vertical (a_h):	0.106773	m
Peso Pistola (Kg):	2.37	Kg

Tabla 34 Datos generales pistola ABIROB A

3.2 TRAYECTORIA 1: ROBOTWORKS

La trayectoria Robotworks (Fig. 36) consta de 225 puntos ubicados en el espacio cartesiano y fue obtenida a partir de dos archivos PanCirc.SRC y PanCirc.DAT disponibles en (Compucraft Ltd., 2012). Estos archivos contienen información de los datos de la herramienta, trayectoria y movimiento de robots marca KUKA y son usados para cargar la trayectoria en los controladores de estos robots. Los archivos pueden ser leídos con un editor de notas por lo que es posible obtener información de la trayectoria y orientación de la herramienta, la cual se extrajo y se validó mediante un análisis de la cinemática inversa de la trayectoria, tanto para el manipulador PUMA 560 y KUKA KR5 HW ARC.

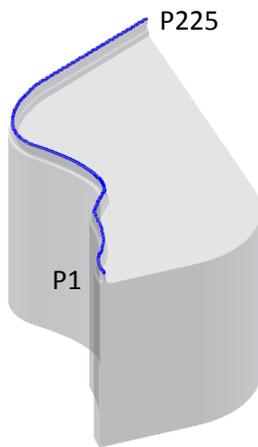


Figura 36 Trayectoria Robotworks

El peso a asignar para cada criterio de optimización (comprendido entre 0 y 1), involucrado en la función de costo a minimizar, fue determinado experimentalmente evaluando el comportamiento del algoritmo en la optimización de las trayectorias Robotworks y Tarro Inclinado, partiendo con un coeficiente de ralentización de 0.3.

En la Fig. 37, se presentan las gráficas de la influencia de los pesos asignados a los criterios de consumo eléctrico y 1-w en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot PUMA 560 con parámetros dinámicos.

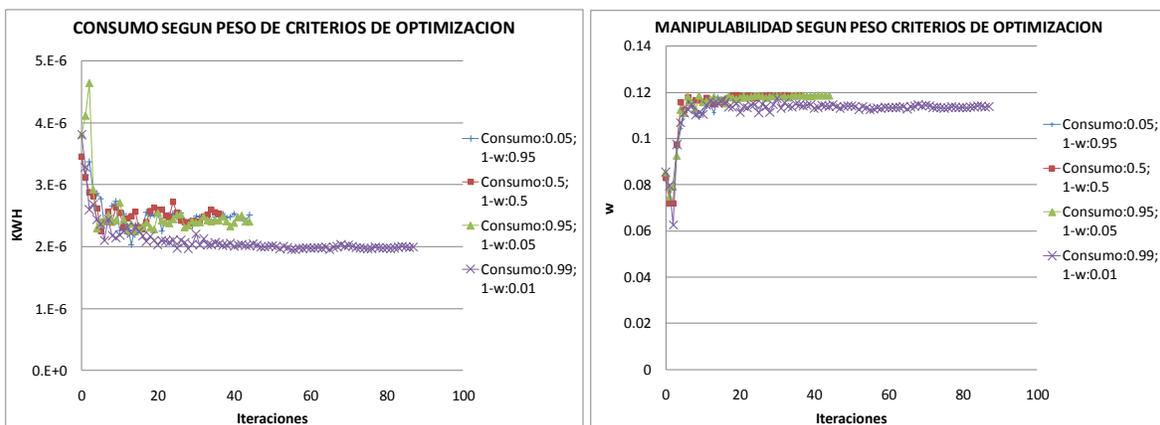


Figura 37 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 con parámetros dinámicos

De la Fig. 37 se evidencia que el algoritmo con un peso de 0.99 en el consumo y 0.01 en 1-w presenta la trayectoria con el menor consumo y la manipulabilidad más baja, y requiere de más iteraciones para cumplir con el criterio de paro. Los algoritmos con los otros tres criterios presentan un comportamiento similar.

En la Fig. 38, se presentan las gráficas de la influencia de los pesos asignados a los criterios de consumo eléctrico y 1-w en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot PUMA 560 solo con parámetros cinemáticos.

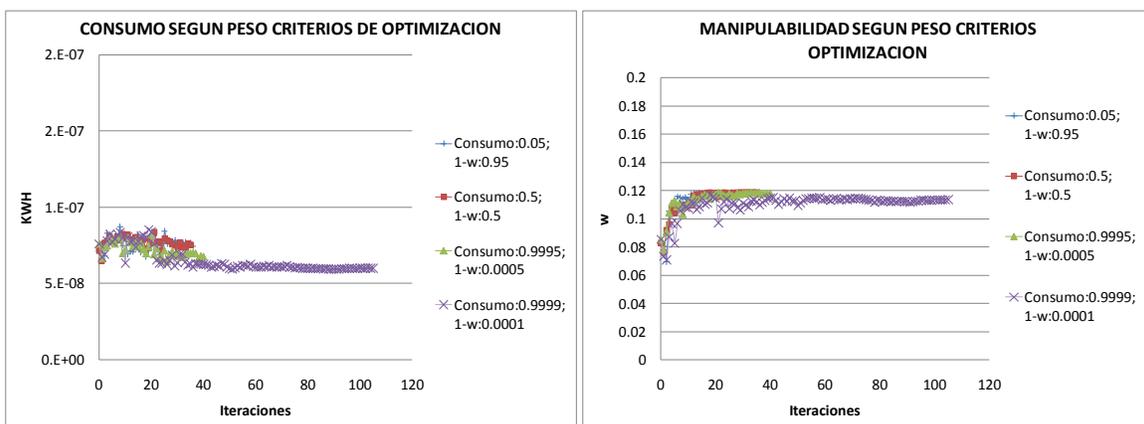


Figura 38 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 38 se evidencia que el algoritmo con un peso de 0.9999 en el consumo y 0.0001 en 1-w presenta la trayectoria con el menor consumo y la manipulabilidad más baja, y requiere de más

iteraciones para cumplir con el criterio de paro. El algoritmo que sigue con menor consumo tiene un peso de 0.9995 en el consumo y 0.0005 en 1-w y requiere un número considerablemente menor de iteraciones para cumplir con el criterio de paro. Los algoritmos con los otros dos criterios presentan un comportamiento similar.

En la Fig. 39, se presentan las gráficas de la influencia de los pesos asignados a los criterios de consumo eléctrico y 1-w en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

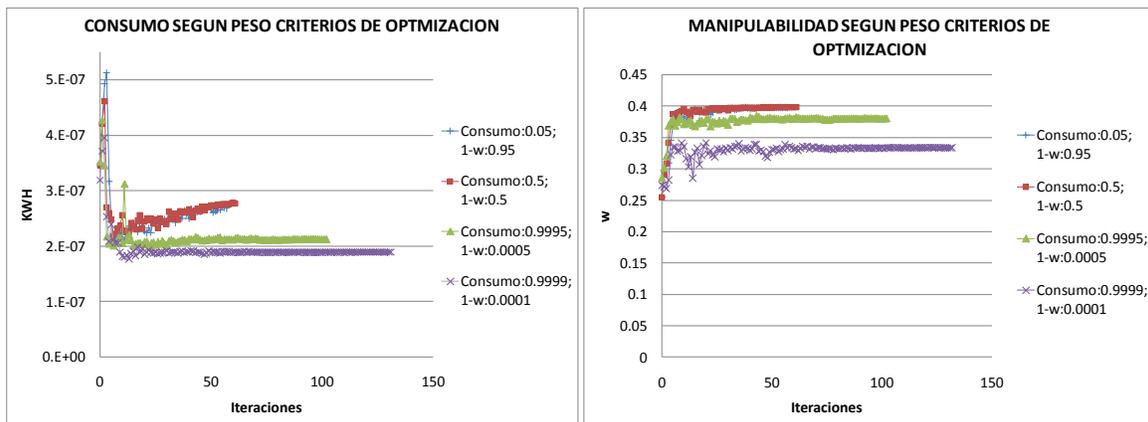


Figura 39 Influencia de pesos criterios optimización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot KUKA KR5 HW ARC solo con parámetros cinemáticos

De la Fig. 39 se evidencia que el algoritmo con un peso de 0.9999 en el consumo y 0.0001 en 1-w presenta la trayectoria con el menor consumo y la manipulabilidad más baja, y requiere de más iteraciones para cumplir con el criterio de paro. El algoritmo que sigue con menor consumo y manipulabilidad mayor que el algoritmo anterior, tiene un peso de 0.9995 en el consumo y 0.0005 en 1-w, con la ventaja que requiere un número menor de iteraciones para cumplir con el criterio de paro. Los algoritmos con los otros dos criterios presentan un comportamiento similar.

En la tabla 35 se hace un resumen comparativo de los resultados obtenidos por el algoritmo de optimización en la trayectoria Robotworks con pesos para los criterios de optimización comprendidos entre 0.05 y 0.99 para el robot PUMA 560 con todos los parámetros dinámicos, y de 0.05 a 0.9999 para los robots PUMA 560 solo con parámetros cinemáticos y KUKA KR5 HW ARC solo con parámetros cinemáticos.

TRAYECTORIA ROBOTWORKS							
ROBOT	Pesos		Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio	
	Consumo	1-w					
PUMA 560 con parámetros dinámicos	0.05	0.95	43	0.88266656	2.4211E-06	0.1187965	
	0.5	0.5	35	0.92200873	2.55985E-06	0.1183293	
	0.95	0.05	43	0.88118057	2.40116E-06	0.1187381	
	0.99	0.01	86	0.82940748	1.98286E-06	0.1136248	
PUMA 560 solo parámetros cinemáticos	0.05	0.95	35	0.91278298	7.67421E-08	0.1186616	
	0.5	0.5	34	0.91760369	7.59396E-08	0.1187616	
	0.9995	0.0005	39	0.86602298	6.7863E-08	0.1181660	
	0.9999	0.0001	104	0.8285272	5.98959E-08	0.1135005	
KUKA KR5 HW ARC solo parámetros cinemáticos	0.05	0.95	56	0.56584022	2.68868E-07	0.3979897	
	0.5	0.5	60	0.56702795	2.7798E-07	0.3984057	
	0.9995	0.0005	101	0.59554978	2.11574E-07	0.3804385	
	0.9999	0.0001	131	0.61697000	1.8906E-07	0.3334293	

Tabla 35 Comparación resultados diferentes pesos en criterios de optimización para trayectoria Robotworks.

De la tabla 35 se observa que los pesos idóneos para los criterios de optimización en manipuladores con todos los parámetros dinámicos son los valores de 0.95 para el consumo eléctrico y 0.05 para la manipulabilidad en robots con todos los parámetros dinámicos, debido a que con estos se plantea un algoritmo que posee la segunda menor cantidad de iteraciones para alcanzar el criterio de paro (43 iteraciones), se obtiene el segundo menor consumo de energía eléctrica en la trayectoria (2.40116E-06 KWH), se logra la segunda manipulabilidad promedio más alta de la trayectoria (0.1187381) y como ventaja adicional posee el segundo menor tiempo de ejecución de la trayectoria (0.88118057 seg).

Igualmente de la tabla 35 se identifican que los pesos idóneos para manipuladores de los cuales solo se conocen los parámetros cinemáticos son de 0.9995 para el consumo eléctrico y 0.0005 para la manipulabilidad, debido a que con estos se plantea un algoritmo que a pesar de poseer la tercer menor cantidad de iteraciones para alcanzar el criterio de paro (39 iteraciones para el PUMA 560 y 101 para KUKA KR5 HW ARC), se obtiene el segundo menor consumo de energía eléctrica en la trayectoria (6.7863E-08 KWH para el PUMA 560 y 2.11574E-07 KWH para KUKA KR5 HW ARC), se logra la tercer manipulabilidad promedio más alta de la trayectoria (0.1181660 para el PUMA 560 y 0.3804385 para KUKA KR5 HW ARC) y como ventaja adicional posee el segundo menor tiempo de ejecución de la trayectoria para el robot PUMA 560 (0.86602298 seg).y el tercer menor tiempo de ejecución de la trayectoria para el robot KUKA KR5 HW ARC(0.59554978 seg).

Estos datos fueron corroborados aplicando el procedimiento anterior en la trayectoria Tarro Inclinado (Tabla 41).

Para definir el coeficiente de ralentización ideal del algoritmo se procedió a evaluar la optimización de la trayectoria Robotworks con los pesos definidos en el párrafo anterior y con valores de coeficientes de ralentización comprendidos entre 0.1 y 1, aplicando el algoritmo en los robots PUMA 560 con parámetros dinámicos, PUMA 560 solo con parámetros cinemáticos, y robot KUKA KR5 HW solo con parámetros cinemáticos.

En la Fig. 40, se presentan las gráficas de la influencia del coeficiente de ralentización en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot PUMA 560 con parámetros dinámicos.

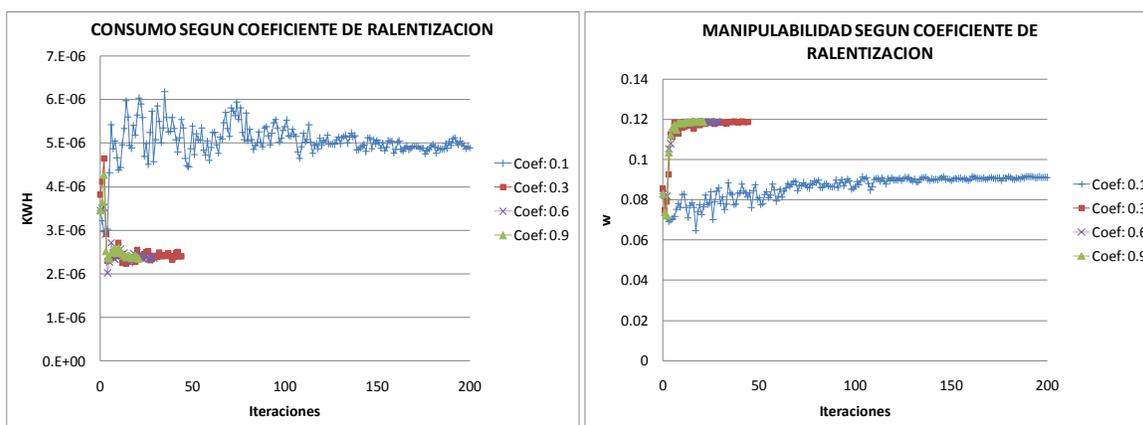


Figura 40 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 con parámetros dinámicos

De la Fig. 40 se evidencia que el algoritmo con un coeficiente de ralentización de 0.1 presenta la trayectoria con el mayor consumo y la menor manipulabilidad, y requiere de más iteraciones para cumplir con el criterio de paro. Los algoritmos con los otros tres coeficientes presentan un comportamiento similar, pero entre éstos destaca el algoritmo con el coeficiente de 0.9 porque requiere la menor cantidad de iteraciones para alcanzar el criterio de paro.

En la Fig. 41, se presentan las gráficas de la influencia del coeficiente de ralentización en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot PUMA 560 solo con parámetros cinemáticos.

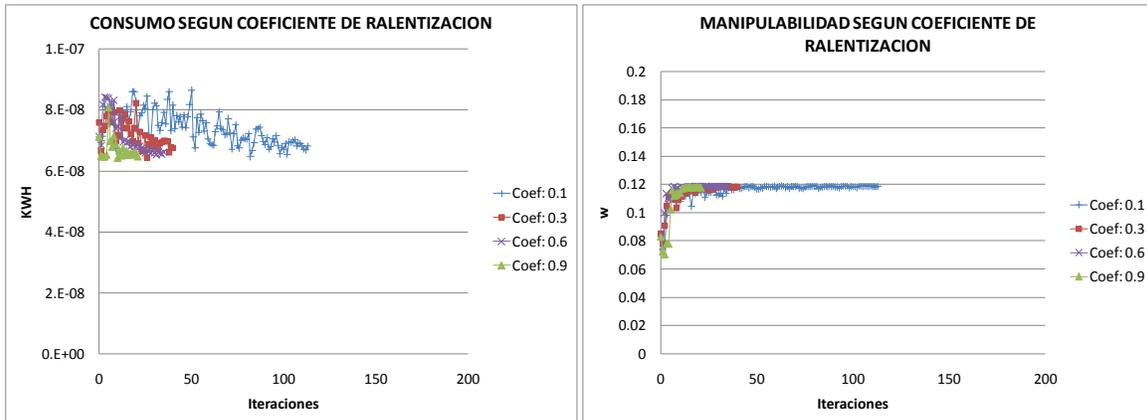


Figura 41 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 41 se evidencia que los cuatro algoritmos presentan un comportamiento similar en la optimización del consumo y de la manipulabilidad, pero aquel que posee un coeficiente de ralentización de 0.1 requiere la mayor cantidad de iteraciones para cumplir con el criterio de paro, y el algoritmo con el coeficiente de 0.9 requiere la menor cantidad de iteraciones para alcanzar el criterio de paro.

En la Fig. 42, se presentan las gráficas de la influencia del coeficiente de ralentización en el comportamiento del consumo eléctrico y manipulabilidad de la trayectoria Robotworks aplicado en el robot KUKA KR5 HW solo con parámetros cinemáticos.

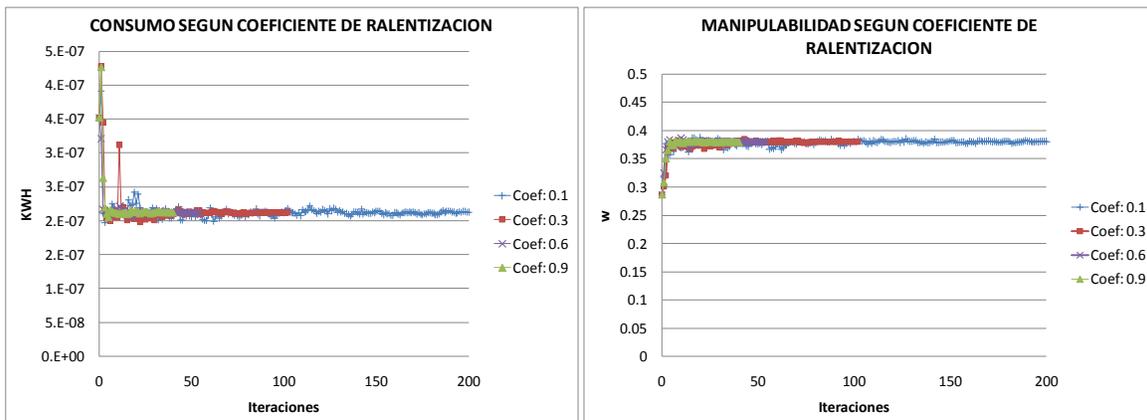


Figura 42 Influencia de coeficiente de ralentización en consumo eléctrico y manipulabilidad de Trayectoria Robotworks en Robot KUKA KR5 HW ARC solo con parámetros cinemáticos

De la Fig. 42 se evidencia que los cuatro algoritmos presentan un comportamiento similar en la optimización del consumo y manipulabilidad, pero aquel que posee un coeficiente de ralentización de 0.1 requiere la mayor cantidad de iteraciones para cumplir con el criterio de paro, y el

algoritmo con el coeficiente de 0.9 requiere la menor cantidad de iteraciones para alcanzar el criterio de paro.

En la tabla 36 se hace un resumen comparativo de los resultados obtenidos por el algoritmo de optimización en la trayectoria Robotworks con valores de coeficiente de ralentización de 0.1, 0.3, 0.6 y 0.9 para los robots: PUMA 560 con todos los parámetros dinámicos, PUMA 560 solo con parámetros cinemáticos y KUKA KR5 HW ARC solo con parámetros cinemáticos.

TRAYECTORIA ROBOTWORKS					
ROBOT	Coeficiente Ralentización	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
PUMA 560 con parámetros dinámicos	0.1	200	1.10372359	4.88738E-06	0.0910493
	0.3	43	0.88118057	2.40116E-06	0.1187381
	0.6	28	0.88599018	2.38425E-06	0.1186974
	0.9	20	0.87839920	2.38541E-06	0.1186826
PUMA 560 solo parámetros cinemáticos	0.1	112	0.87014051	6.68615E-08	0.1183898
	0.3	39	0.86602298	6.7863E-08	0.1181660
	0.6	34	0.87187156	6.55868E-08	0.1184741
	0.9	20	0.87179850	6.58081E-08	0.1184108
KUKA KR5 HW ARC solo parámetros cinemáticos	0.1	200	0.59703184	2.12713E-07	0.3808823
	0.3	101	0.59554978	2.11574E-07	0.3804385
	0.6	53	0.59661045	2.11522E-07	0.3801365
	0.9	39	0.59706181	2.11951E-07	0.3803341

Tabla 36 Comparación resultados de coeficientes de ralentización en trayectoria Robotworks.

De la tabla 36 se puede concluir que a pesar de obtener valores optimizados similares en los tres manipuladores con los cuatro algoritmos, el valor idóneo del coeficiente de ralentización a usar en el algoritmo de optimización tanto para manipuladores con todos los parámetros dinámicos y para manipuladores con solo los parámetros cinemáticos es de 0.9 debido a que con este coeficiente se requiere la menor cantidad de iteraciones para cumplir la regla de paro del algoritmo (20 iteraciones para el PUMA 560 con todos los parámetros dinámicos, 20 iteraciones para el PUMA 560 con solo los parámetros cinemáticos, y 39 iteraciones para el KUKA KR5 HW ARC con solo los parámetros cinemáticos. Este dato fue corroborado aplicando el procedimiento anterior en la trayectoria Tarro Inclinado (Tabla 42).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Robotworks y se compara con la trayectoria sin optimizar, en el robot PUMA 560 con todos los parámetros dinámicos.

En la Fig. 43 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico a medida que el algoritmo optimiza la trayectoria Robotworks en cada iteración, realizada por el robot PUMA 560 con todos los parámetros dinámicos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 20.

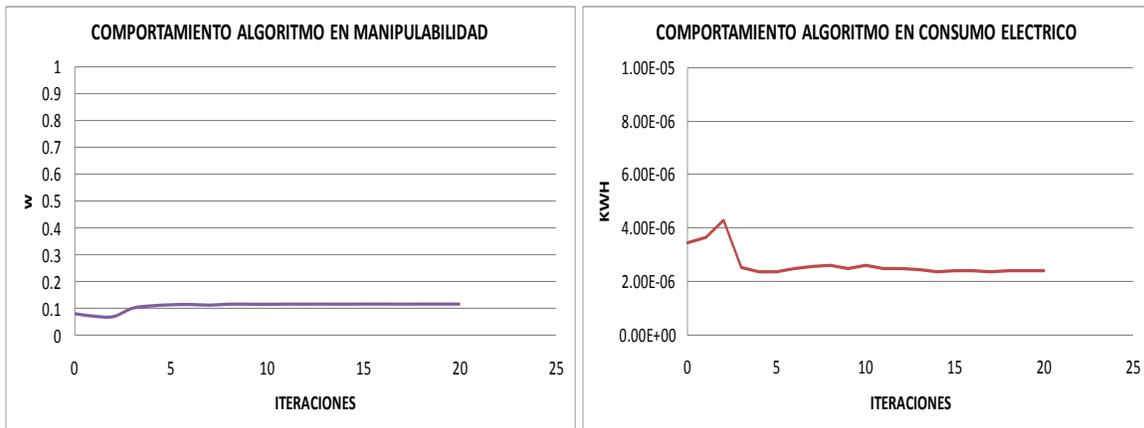


Figura 43 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 44 se identifican los recorridos que deben hacer las articulaciones del robot PUMA 560 con parámetros dinámicos y solo con parámetros cinemáticos, para seguir la trayectoria Robotworks.

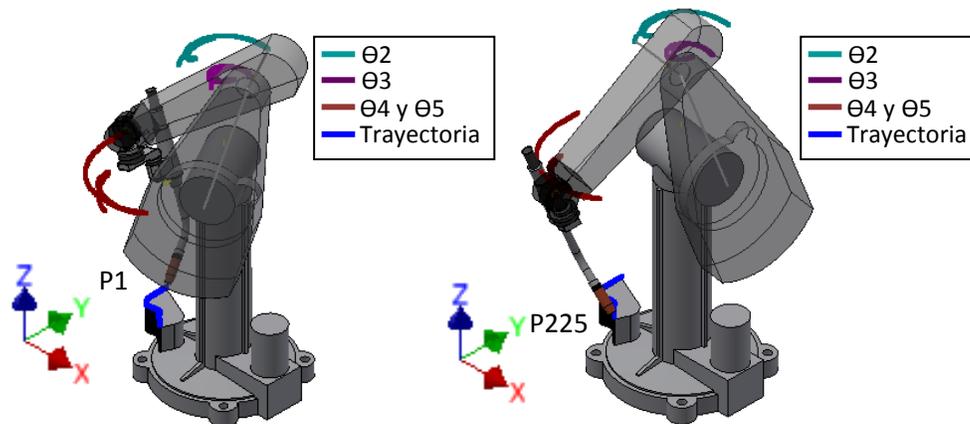


Figura 44 Trayectoria Robotworks en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos

En la Fig. 45 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 con todos los parámetros dinámicos, para seguir la trayectoria optimizada Robotworks. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.61642617 m, Angulo R: 0.51938117 rad y Angulo P: -2.34115865 rad.

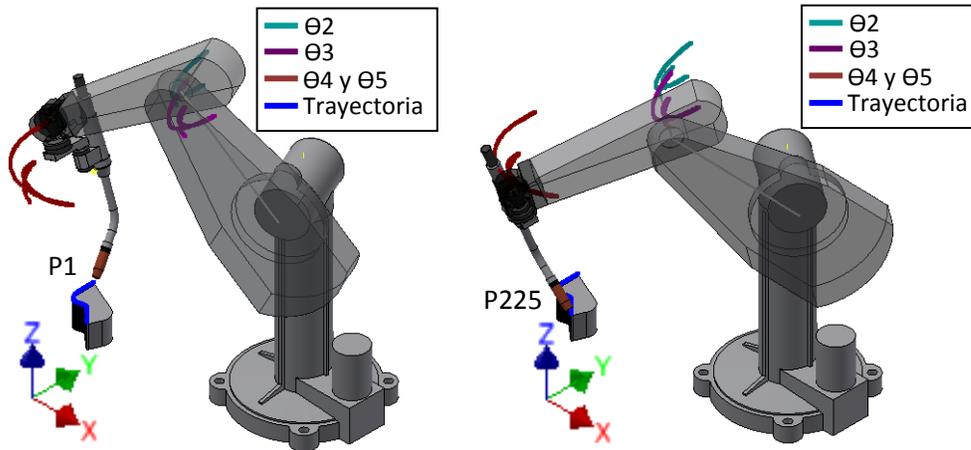


Figura 45 Trayectoria Robotworks Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 46 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Robotworks realizada por el robot PUMA 560 con todos los parámetros dinámicos.

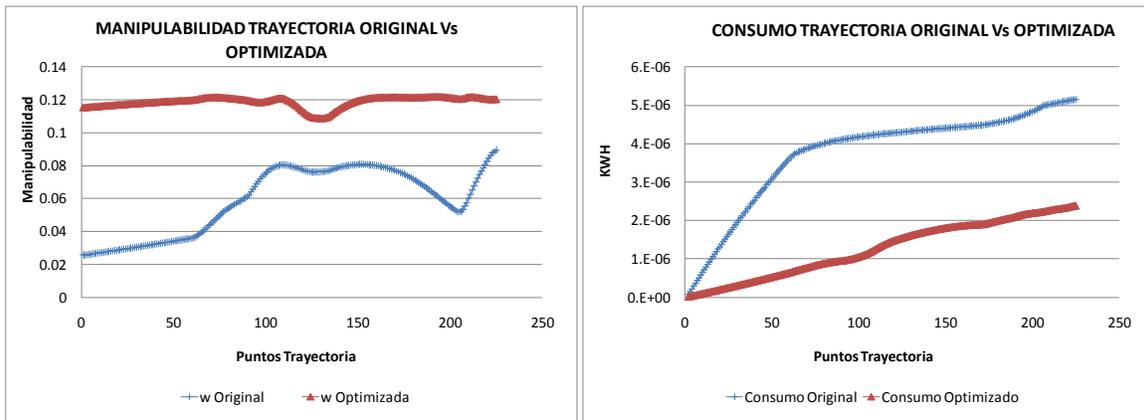


Figura 46 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original Vs optimizada en Robot PUMA 560 con parámetros dinámicos

De la Fig. 46 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la

trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 37 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Robotworks original y optimizada del robot PUMA 560 con parámetros dinámicos, evidenciándose que el algoritmo de optimización en la iteración 20 logra optimizaciones del 53.73% en el consumo eléctrico, el 50.38% en la manipulabilidad, incluso del 13.21% en el tiempo de la trayectoria.

TRAYECTORIA ROBOTWORKS EN PUMA 560 CON PARAMETROS DINAMICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.01205355	5.15545E-06	0.05889245
Trayectoria Optimizada	20	0.87839920	2.38541E-06	0.11868260
% Optimización		13.21%	53.73%	50.38%

Tabla 37 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot PUMA 560 con parámetros dinámicos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 37 se puede identificar la productividad para la trayectoria original en $1.94E+05$ unidades/KWH, y para la trayectoria optimizada en $4.19E+05$ unidades/KWH, obteniendo un aumento en la productividad del 53.73%.

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Robotworks y se compara con la trayectoria sin optimizar, en el robot PUMA 560 solo con parámetros cinemáticos.

En la Fig. 47 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Robotworks en cada iteración, realizada por el robot PUMA 560 solo con parámetros cinemáticos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 20.

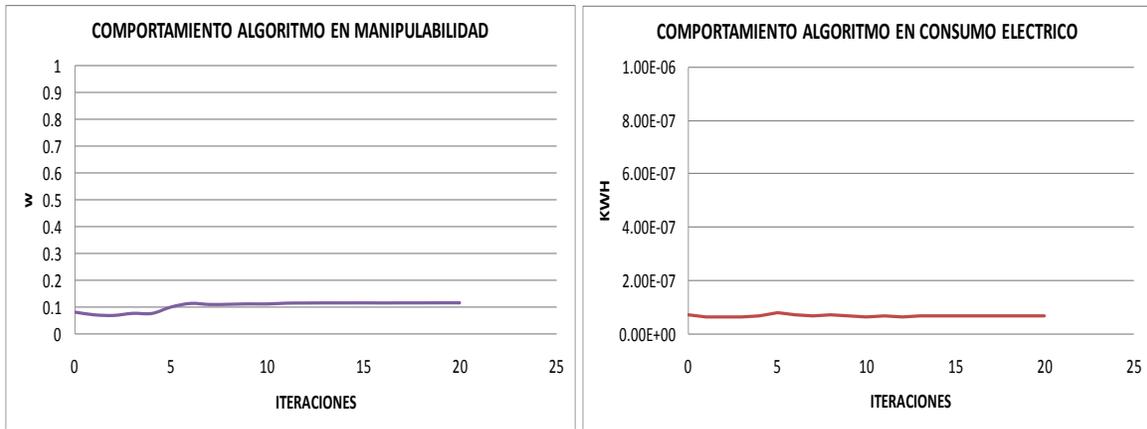


Figura 47 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 48 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 solo con parámetros cinemáticos para seguir la trayectoria optimizada Robotworks. Para obtener esta trayectoria los datos del espacio de trabajo simplificado usados para definir el punto inicial son: Radio = 0.4940434 m, Angulo R: 0.4940434 rad y Angulo P: -2.41708055 rad.

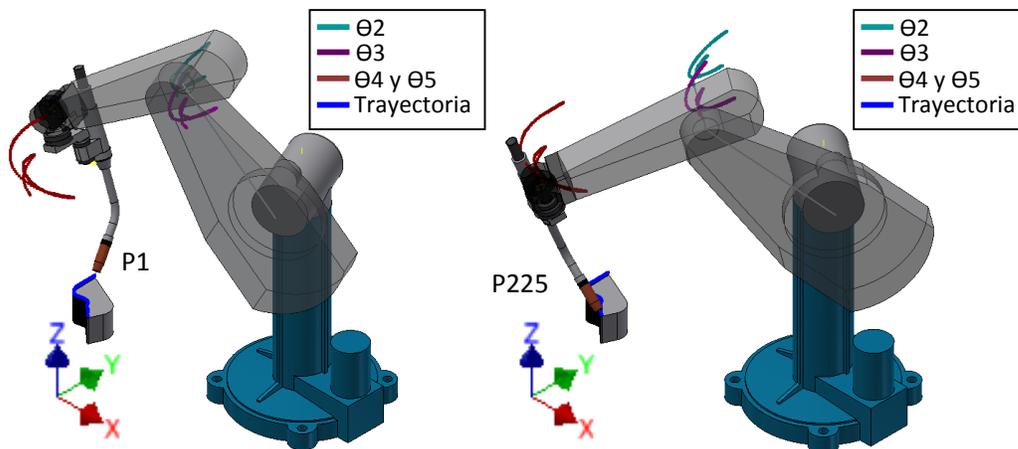


Figura 48 Trayectoria Robotworks Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 49 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Robotworks realizada por el robot PUMA 560 solo con parámetros cinemáticos.

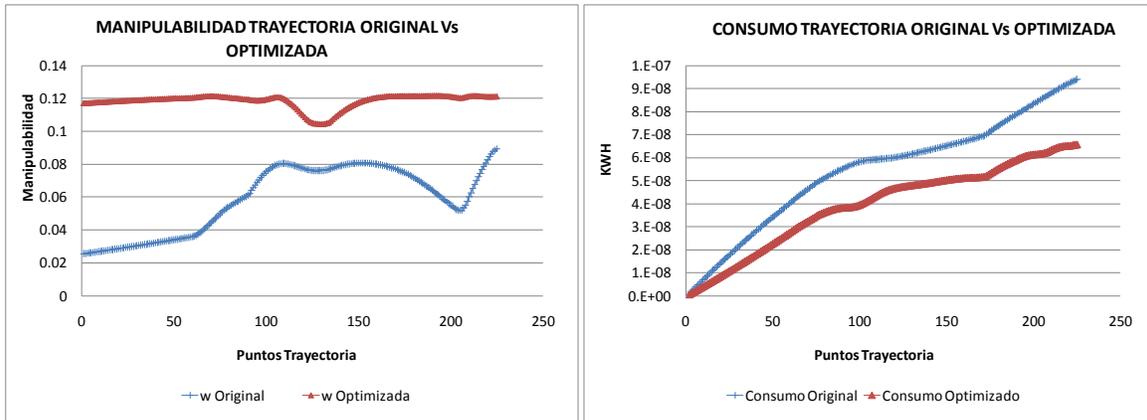


Figura 49 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 49 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 38 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Robotworks original y optimizada del robot PUMA 560 con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 20 logra optimizaciones del 30.02% en el consumo eléctrico, el 50.26% en la manipulabilidad, incluso del 13.86% en el tiempo de la trayectoria.

TRAYECTORIA ROBOTWORKS EN PUMA 560 CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	1.01205355	9.40424E-08	0.05889245
Trayectoria Optimizada	20	0.87179850	6.58081E-08	0.11841078
% Optimización		13.86%	30.02%	50.26%

Tabla 38 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 38 se puede identificar la productividad para la trayectoria original en 1.06E+07 unidades/KWH, y para

la trayectoria optimizada en $1.52E+07$ unidades/KWH, obteniendo un aumento en la productividad del 30.02%.

Con los datos del espacio de trabajo simplificado que generaron esta trayectoria, se procede a evaluar su comportamiento en el robot PUMA 560 con todos los parámetros dinámicos, la cual permite identificar que tan eficiente son estos datos de ubicación de la trayectoria en el espacio de trabajo en la optimización de los parámetros de consumo eléctrico y manipulabilidad. Estos resultados se muestran en la siguiente tabla:

TRAYECTORIA ROBOTWORKS EN PUMA 560 CON PARAMETROS DINAMICOS			
ROBOT	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original Parámetros dinámicos	1.01205355	5.16E-06	0.05889245
Trayectoria Evaluada	0.87179850	2.38298E-06	0.11841078
% Optimización	13.86%	53.78%	50.26%

Tabla 39 Comparación de parámetros optimizados en trayectoria Robotworks original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos

De la tabla 39 se evidencia que la trayectoria obtenida con el algoritmo de optimización en el robot PUMA 560 con solo parámetros cinemáticos, genera optimizaciones en el robot con todos los parámetros dinámicos muy satisfactorias: 53.78% en el consumo eléctrico y 50.26% en la manipulabilidad. Estos resultados son similares a los obtenidos al aplicar el algoritmo para optimizar esta trayectoria en el robot PUMA 560 con todos los parámetros dinámicos (tabla 37).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Robotworks y se compara con la trayectoria sin optimizar, en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

En la figura 50 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Robotworks en cada iteración, realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos. En esta se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 39.

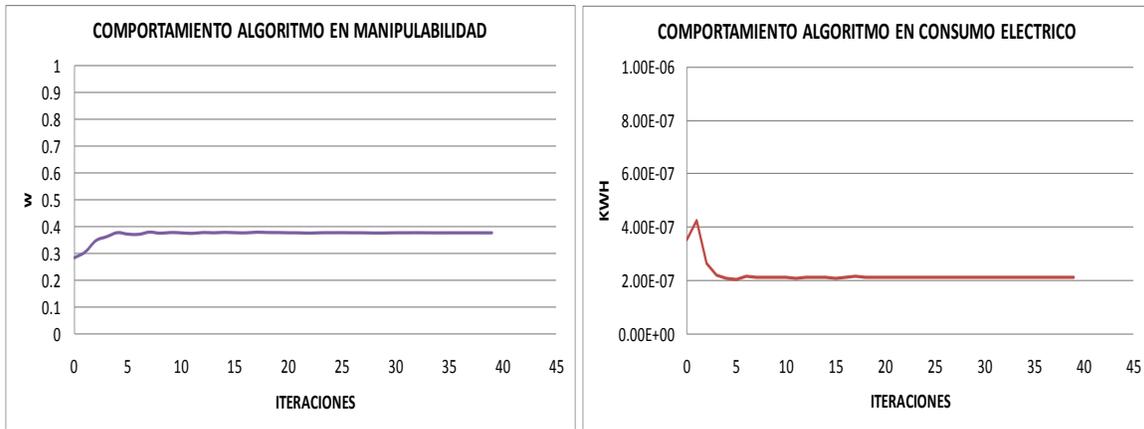


Figura 50 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Robotworks Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 51 se identifican los recorridos que deben hacer las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria Robotworks.

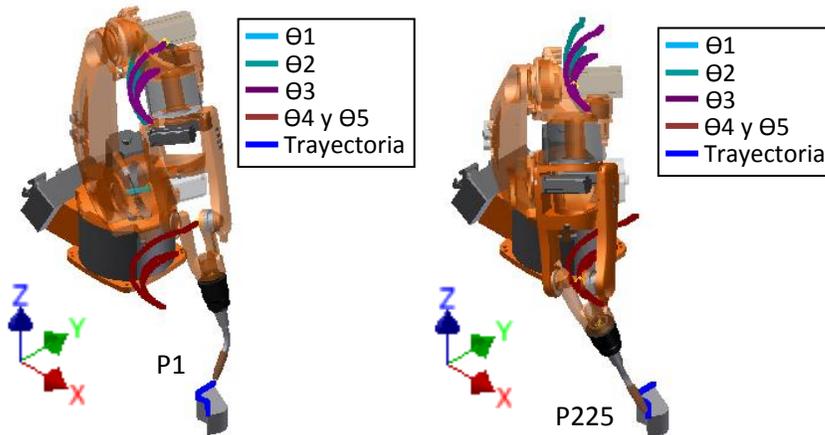


Figura 51 Trayectoria Robotworks en Robot KUKA KR5 HW

En la Fig. 52 se grafican los recorridos que deben realizar las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria optimizada Robotworks. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.90076568 m, Angulo R: 0.15411214 rad y Angulo P: -2.0967176 rad.

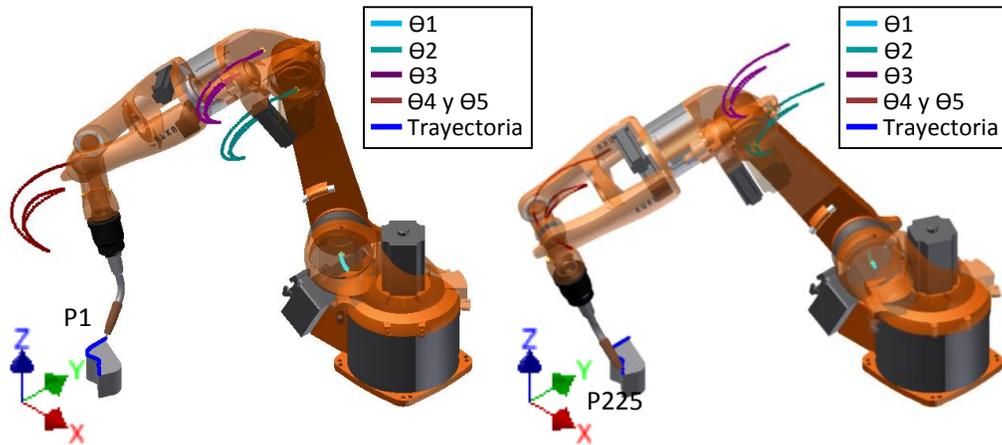


Figura 52 Trayectoria Robotworks Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 53 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Robotworks realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

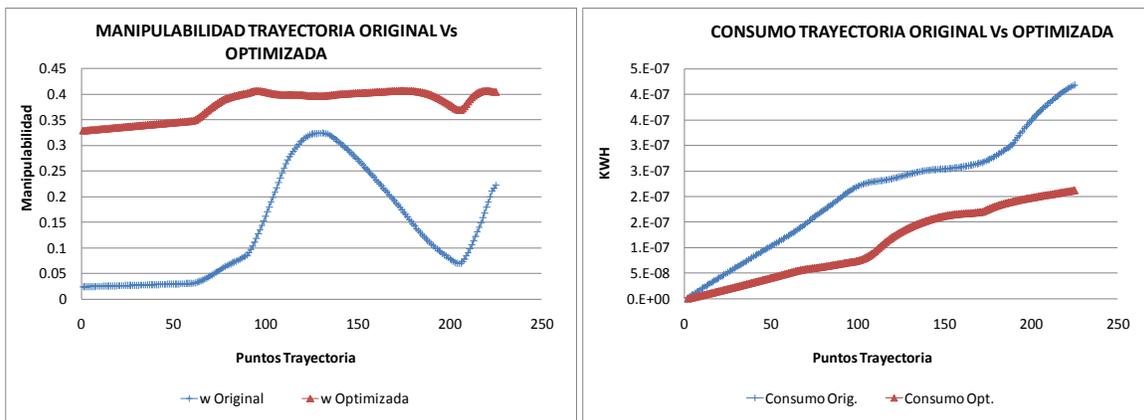


Figura 53 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Robotworks original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

De la Fig. 53 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 40 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Robotworks original y optimizada del robot KUKA KR5 HW ARC con solo

parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 39 logra optimizaciones del 49.34% en el consumo eléctrico, el 64.40% en la manipulabilidad, incluso del 12.24% en el tiempo de la trayectoria.

TRAYECTORIA ROBOTWORKS EN KUKA KR5 HW ARC CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	0.68033043	4.18358E-07	0.13538526
Trayectoria Optimizada	39	0.59706181	2.11951E-07	0.3803341
% Optimización		12.24%	49.34%	64.40%

Tabla 40 Comparación de parámetros optimizados en trayectoria Robotworks original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 40 se puede identificar la productividad para la trayectoria original en $2.39E+06$ unidades/KWH, y para la trayectoria optimizada en $4.72E+06$ unidades/KWH, obteniendo un aumento en la productividad del 49.34%.

3.3 TRAYECTORIA 2: TARRO INCLINADO

Consta de 413 puntos y fue obtenida a partir del espacio cartesiano mediante un algoritmo de extracción de líneas 3D identificadas en un sólido modelado en Autodesk Inventor 2013® en conjunto con una orientación deseada validada por medio un análisis de la cinemática inversa de la trayectoria (Fig. 54).

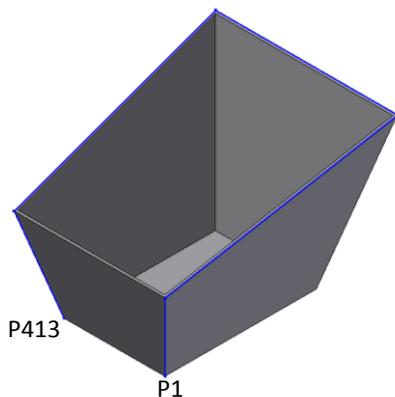


Figura 54 Trayectoria Tarro Inclinado

El peso a asignar para cada criterio de optimización (comprendido entre 0 y 1), involucrado en la función de costo a minimizar, fue determinado experimentalmente evaluando el comportamiento del algoritmo en la optimización de las trayectorias Robotworks y Tarro Inclinado, partiendo con un coeficiente de ralentización de 0.3.

En la tabla 41 se hace un resumen comparativo de los resultados obtenidos por el algoritmo de optimización en la trayectoria Tarro Inclinado con pesos para los criterios de optimización comprendidos entre 0.05 y 0.99 para el robot PUMA 560 con todos los parámetros dinámicos, y de 0.05 a 0.9999 para los robots PUMA 560 solo con parámetros cinemáticos y KUKA KR5 HW ARC solo con parámetros cinemáticos.

TRAYECTORIA TARRO INCLINADO						
ROBOT	Pesos		Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
	Consumo	1-w				
PUMA 560 con parámetros dinámicos	0.05	0.95	169	1.66593414	2.16343E-05	0.1124357
	0.5	0.5	188	1.65632152	2.24279E-05	0.1111747
	0.95	0.05	81	1.65655674	1.57009E-05	0.1061946
	0.99	0.01	64	1.65860324	1.40759E-05	0.0930911
PUMA 560 solo parámetros cinemáticos	0.05	0.95	61	1.66588926	4.67389E-07	0.1121672
	0.5	0.5	63	1.66442514	4.44439E-07	0.1121535
	0.9995	0.0005	58	1.65729157	6.40324E-08	0.1040066
	0.9999	0.0001	51	1.65919588	6.29201E-08	0.1013339
KUKA KR5 HW ARC solo parámetros cinemáticos	0.05	0.95	55	1.65692232	2.97891E-06	0.3934071
	0.5	0.5	56	1.65728362	2.87552E-06	0.3931454
	0.9995	0.0005	63	1.65757677	5.39091E-07	0.3157752
	0.9999	0.0001	90	1.656497	4.67788E-07	0.2778002

Tabla 41 Comparación resultados diferentes pesos en criterios de optimización para trayectoria Tarro Inclinado.

De la tabla 41 se observa que los pesos idóneos para los criterios de optimización en manipuladores con todos los parámetros dinámicos son los valores de 0.95 para el consumo eléctrico y 0.05 para la manipulabilidad en robots con todos los parámetros dinámicos, debido a que con estos se plantea un algoritmo que posee la segunda menor cantidad de iteraciones para alcanzar el criterio de paro (81 iteraciones), se obtiene el segundo menor consumo de energía eléctrica en la trayectoria (1.57009E-05 KWH), se logra la tercer manipulabilidad promedio más alta de la trayectoria (0.1061946) y como ventaja adicional posee el segundo menor tiempo de ejecución de la trayectoria (1.65655674 seg).

Igualmente de la tabla 41 se identifican que los pesos idóneos para manipuladores de los cuales solo se conocen los parámetros cinemáticos son de 0.9995 para el consumo eléctrico y 0.0005 para la manipulabilidad, debido a que con estos se plantea un algoritmo que a pesar de poseer la segunda menor cantidad de iteraciones para el PUMA 560 (58 iteraciones), y la tercer menor

cantidad de iteraciones en el robot KUKA KR5 HW ARC (63 iteraciones), necesarias para alcanzar el criterio de paro, se obtiene el segundo menor consumo de energía eléctrica en la trayectoria (6.40324E-08 KWH para el PUMA 560 y 5.39091E-07 KWH para KUKA KR5 HW ARC), se logra la tercer manipulabilidad promedio más alta de la trayectoria (0.1040066 para el PUMA 560 y 0.3157752 para KUKA KR5 HW ARC) y como ventaja adicional posee el primer menor tiempo de ejecución de la trayectoria para el robot PUMA 560 (1.65729157 seg).y el tiempo más alto en la ejecución de la trayectoria para el robot KUKA KR5 HW ARC(1.65757677 seg).

Para definir el coeficiente de ralentización ideal del algoritmo se procedió a evaluar la optimización de la trayectoria Tarro Inclinado con los pesos definidos en el párrafo anterior y con valores de coeficientes de ralentización comprendidos entre 0.1 y 1, aplicando el algoritmo en los robots PUMA 560 con parámetros dinámicos, PUMA 560 solo con parámetros cinemáticos, y robot KUKA KR5 HW solo con parámetros cinemáticos.

En la tabla 42 se hace un resumen comparativo de los resultados obtenidos por el algoritmo de optimización en la trayectoria Tarro Inclinado con valores de coeficiente de ralentización de 0.1, 0.3, 0.6 y 0.9 para los robots: PUMA 560 con todos los parámetros dinámicos, PUMA 560 solo con parámetros cinemáticos y KUKA KR5 HW ARC solo con parámetros cinemáticos.

TRAYECTORIA TARRO INCLINADO					
ROBOT	Coeficiente Ralentización	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
PUMA 560 con parámetros dinámicos	0.1	200	1.65635134	1.59187E-05	0.1068016
	0.3	81	1.65655674	1.57009E-05	0.1061946
	0.6	47	1.65650967	1.5727E-05	0.1062944
	0.9	27	1.65655682	1.5657E-05	0.1060444
PUMA 560 solo parámetros cinemáticos	0.1	189	1.65770318	6.32603E-08	0.1037037
	0.3	58	1.65729157	6.40324E-08	0.1040066
	0.6	31	1.65755160	6.36811E-08	0.1039765
	0.9	28	1.65728197	6.32587E-08	0.1040561
KUKA KR5 HW ARC solo parámetros cinemáticos	0.1	200	1.65767840	5.35291E-07	0.3147679
	0.3	63	1.65757677	5.39091E-07	0.3157752
	0.6	40	1.65756994	5.36812E-07	0.3152017
	0.9	32	1.65755754	5.36883E-07	0.3152212

Tabla 42 Comparación resultados de coeficientes de ralentización en trayectoria Tarro Inclinado.

De la tabla 42 se puede concluir que a pesar de obtener valores optimizados similares en los tres manipuladores con los cuatro algoritmos, el valor idóneo del coeficiente de ralentización a usar en el algoritmo de optimización tanto para manipuladores con todos los parámetros dinámicos y para manipuladores con solo los parámetros cinemáticos es de 0.9 debido a que con este coeficiente se requiere la menor cantidad de iteraciones para cumplir la regla de paro del algoritmo (27 iteraciones para el PUMA 560 con todos los parámetros dinámicos, 28 iteraciones para el PUMA 560 con solo los parámetros cinemáticos, y 32 iteraciones para el KUKA KR5 HW ARC con solo los parámetros cinemáticos).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Tarro Inclinado y se compara con la trayectoria sin optimizar, en el robot PUMA 560 con todos los parámetros dinámicos.

En la Fig. 55 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico a medida que el algoritmo optimiza la trayectoria Tarro Inclinado en cada iteración, realizada por el robot PUMA 560 con todos los parámetros dinámicos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 27.

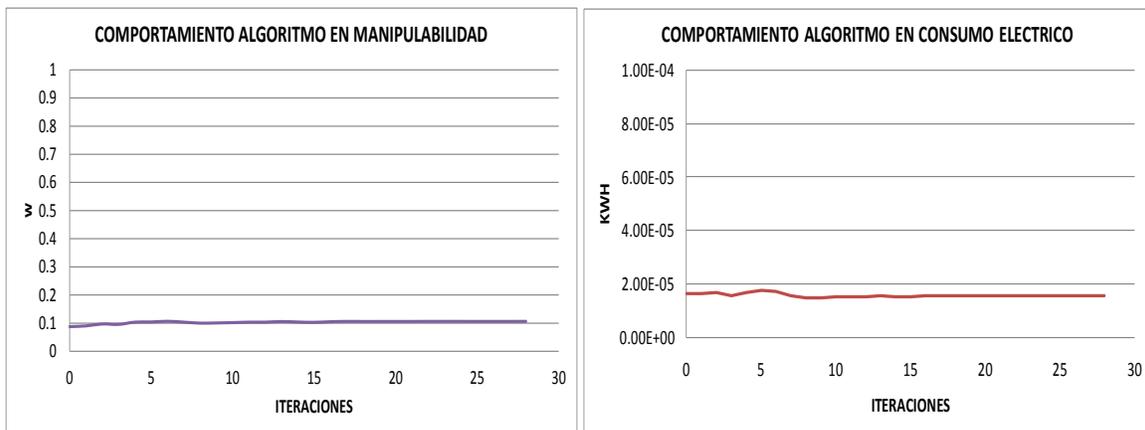


Figura 55 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 56 se identifican los recorridos que deben hacer las articulaciones del robot PUMA 560 con parámetros dinámicos y solo con parámetros cinemáticos, para seguir la trayectoria Tarro Inclinado.

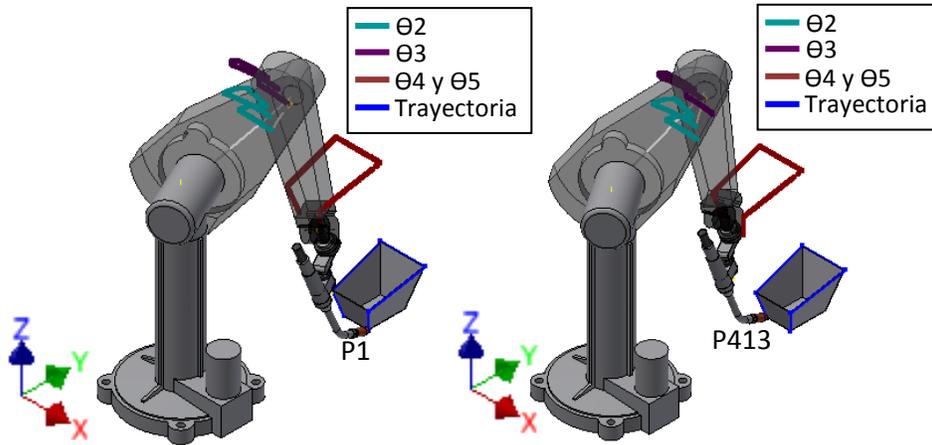


Figura 56 Trayectoria Tarro Inclinado en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos

En la Fig. 57 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 con todos los parámetros dinámicos, para seguir la trayectoria optimizada Tarro Inclinado. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.77493786 m, Angulo R: 0.438154 rad y Angulo P: - 0.05930750 rad.

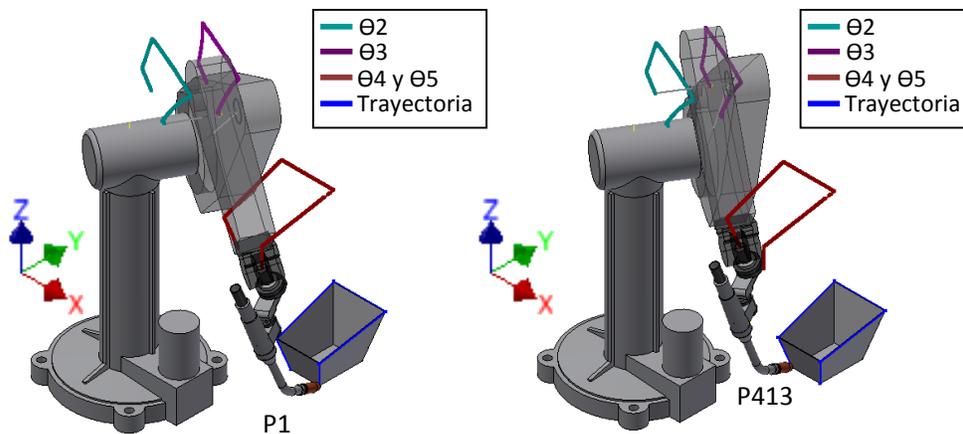


Figura 57 Trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 58 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Tarro Inclinado realizada por el robot PUMA 560 con todos los parámetros dinámicos.

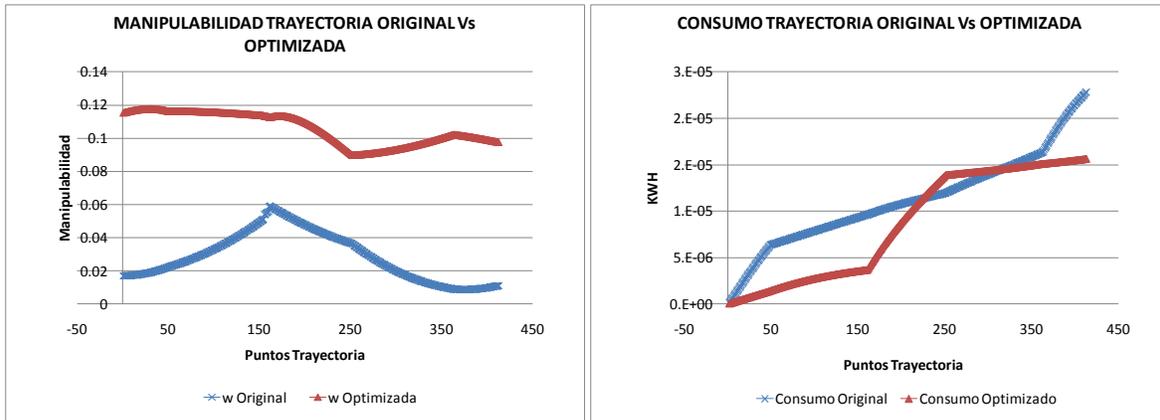


Figura 58 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot PUMA 560 con parámetros dinámicos

De la Fig. 58 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico final de la trayectoria optimizada es menor comparado con el consumo de la trayectoria original. Sin embargo en el punto 225 de la trayectoria original el consumo de la trayectoria original tendía a ser menor, pero nuevamente en el punto 317 de la trayectoria original el consumo comienza a aumentar superando el de la trayectoria optimizada.

En la tabla 43 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Tarro Inclinado original y optimizada del robot PUMA 560 con parámetros dinámicos, evidenciándose que el algoritmo de optimización en la iteración 27 logra optimizaciones del 31.48% en el consumo eléctrico, el 73.02% en la manipulabilidad, incluso del 0.74% en el tiempo de la trayectoria.

TRAYECTORIA TARRO INCLINADO EN PUMA 560 CON PARAMETROS DINAMICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.66888271	2.28492E-05	0.02860888
Trayectoria Optimizada	27	1.65655682	1.5657E-05	0.1060444
% Optimización		0.74%	31.48%	73.02%

Tabla 43 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot PUMA 560 con parámetros dinámicos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 43 se puede identificar la productividad para la trayectoria original en $4.38E+04$ unidades/KWH, y para la trayectoria optimizada en $6.39E+04$ unidades/KWH, obteniendo un aumento en la productividad del 31.48%.

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Tarro Inclinado y se compara con la trayectoria sin optimizar, en el robot PUMA 560 solo con parámetros cinemáticos.

En la Fig. 59 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Tarro Inclinado en cada iteración, realizada por el robot PUMA 560 solo con parámetros cinemáticos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 28.

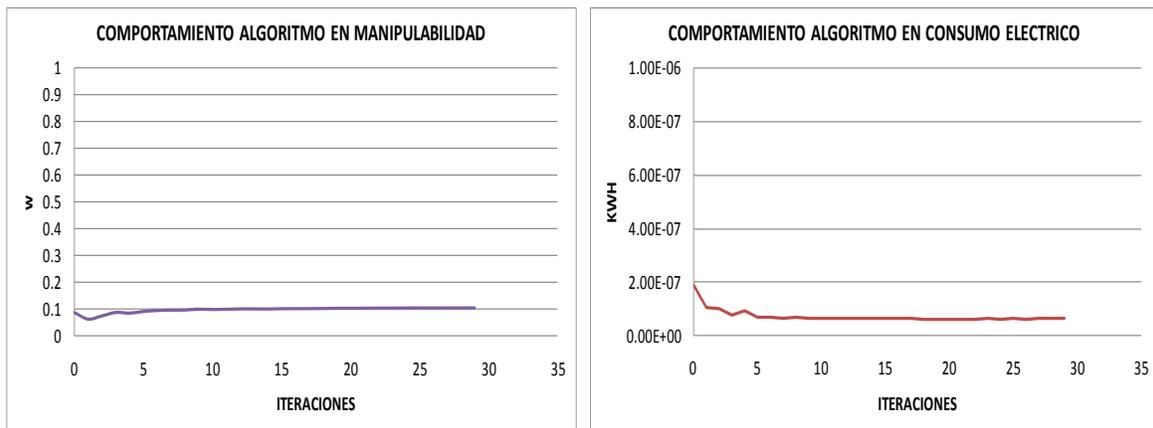


Figura 59 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 60 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 solo con parámetros cinemáticos para seguir la trayectoria optimizada Tarro Inclinado. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.73685633 m, Angulo R: 0.3664736 rad y Angulo P: 0.0253693 rad.

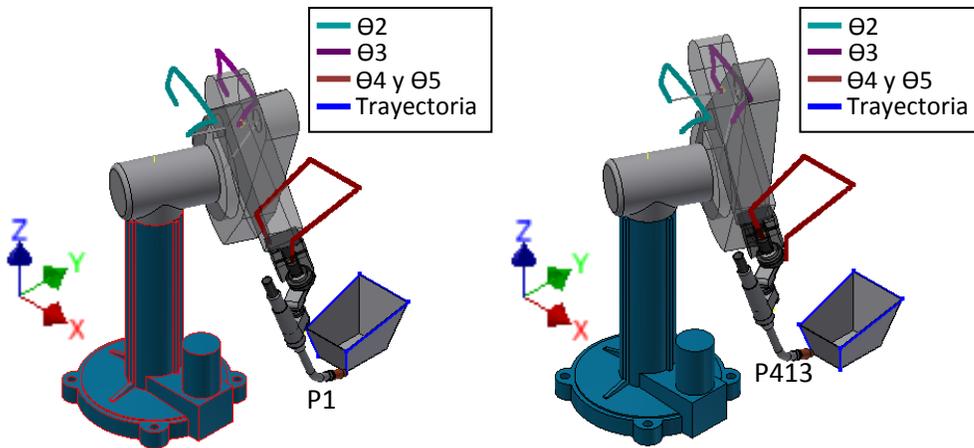


Figura 60 Trayectoria Tarro Inclinado Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 61 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Tarro Inclinado realizada por el robot PUMA 560 solo con parámetros cinemáticos.

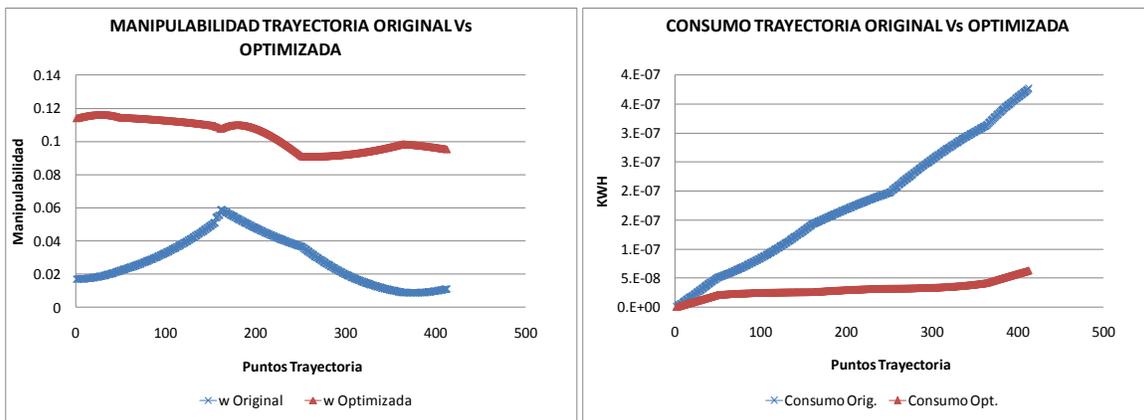


Figura 61 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 61 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 44 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Tarro Inclinado original y optimizada del robot PUMA 560 con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 28 logra optimizaciones del 83.22% en el consumo eléctrico, el 72.51% en la manipulabilidad, incluso del 0.70% en el tiempo de la trayectoria.

TRAYECTORIA TARRO INCLINADO EN PUMA 560 CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	1.66888271	3.76998E-07	0.02860888
Trayectoria Optimizada	28	1.65728197	6.32587E-08	0.1040561
% Optimización		0.70%	83.22%	72.51%

Tabla 44 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 44 se puede identificar la productividad para la trayectoria original en $2.65E+06$ unidades/KWH, y para la trayectoria optimizada en $1.58E+07$ unidades/KWH, obteniendo un aumento en la productividad del 83.22%.

Con los datos del espacio de trabajo simplificado que generaron esta trayectoria, se procede a evaluar su comportamiento en el robot PUMA 560 con todos los parámetros dinámicos, la cual permite identificar que tan eficiente son estos datos de ubicación de la trayectoria en el espacio de trabajo en la optimización de los parámetros de consumo eléctrico y manipulabilidad. Estos resultados se muestran en la siguiente tabla:

TRAYECTORIA TARRO INCLINADO EN PUMA 560 CON PARAMETROS DINAMICOS			
ROBOT	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original Parámetros dinámicos	1.66888271	2.28492E-05	0.02860888
Trayectoria Evaluada	1.65728197	1.54203E-05	0.1040561
% Optimización		0.70%	32.51%
			72.51%

Tabla 45 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos

De la tabla 45 se evidencia que la trayectoria obtenida con el algoritmo de optimización en el robot PUMA 560 con solo parámetros cinemáticos, genera optimizaciones en el robot con todos los parámetros dinámicos muy satisfactorias: 32.51% en el consumo eléctrico y 72.51% en la manipulabilidad. Estos resultados son similares a los obtenidos al aplicar el algoritmo para optimizar esta trayectoria en el robot PUMA 560 con todos los parámetros dinámicos (tabla 43).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Tarro Inclinado y se compara con la trayectoria sin optimizar, en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

En la Fig. 62 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Tarro Inclinado en cada iteración, realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos. En esta se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 32.

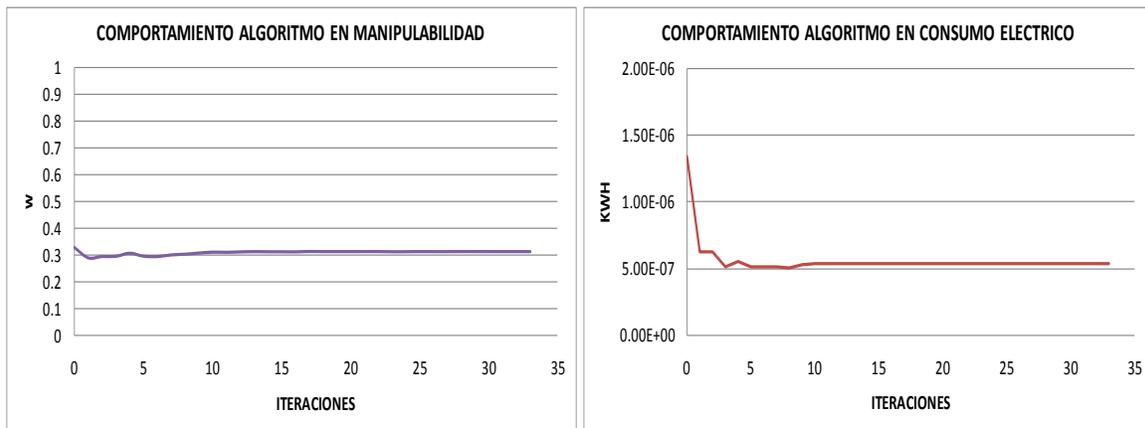


Figura 62 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Tarro Inclinado Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 63 se identifican los recorridos que deben hacer las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria Tarro Inclinado.

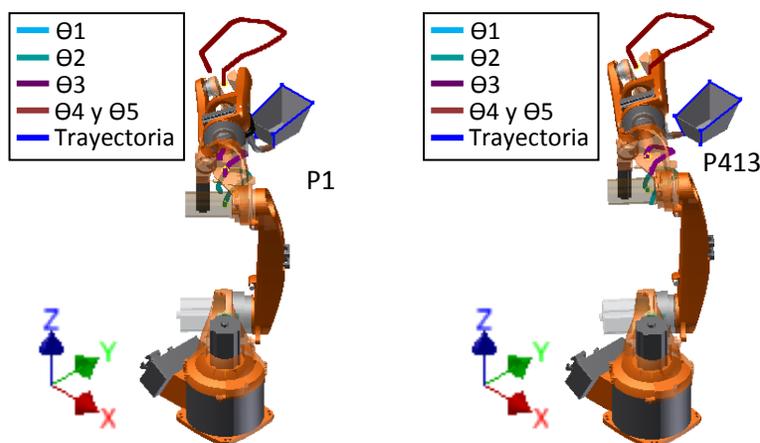


Figura 63 Trayectoria Tarro Inclinado sin optimizar en Robot KUKA KR5 HW

En la Fig. 64 se grafican los recorridos que deben realizar las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos para seguir la trayectoria optimizada Tarro Inclinado. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 1.02095456 m, Angulo R: -0.0342686 rad y Angulo P: 0.4381580 rad.

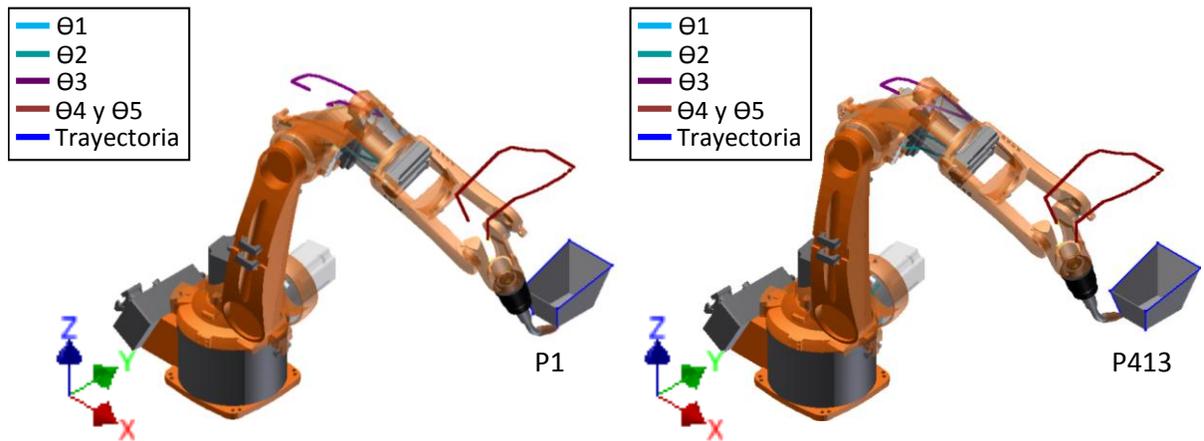


Figura 64 Trayectoria Tarro Inclinado Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 65 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Tarro Inclinado realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

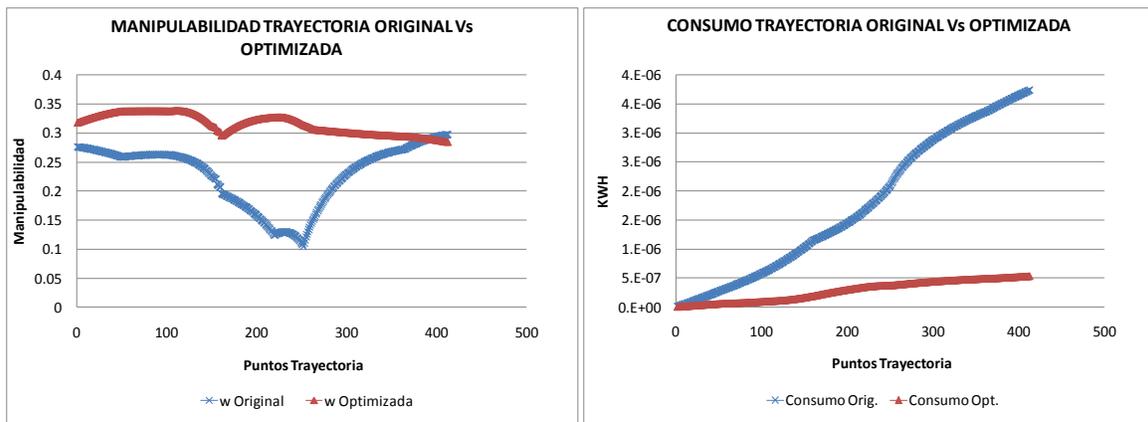


Figura 65 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Tarro Inclinado original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

De la Fig. 65 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la

trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 46 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Tarro Inclinado original y optimizada del robot KUKA KR5 HW ARC con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 32 logra optimizaciones del 85.65% en el consumo eléctrico, el 27.45% en la manipulabilidad, incluso del 0.44% en el tiempo de la trayectoria.

TRAYECTORIA TARRO INCLINADO EN KUKA KR5 HW ARC CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.66491957	3.74087E-06	0.22870419
Trayectoria Optimizada	32	1.65755754	5.36883E-07	0.3152212
% Optimización		0.44%	85.65%	27.45%

Tabla 46 Comparación de parámetros optimizados en trayectoria Tarro Inclinado original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 46 se puede identificar la productividad para la trayectoria original en $2.67E+05$ unidades/KWH, y para la trayectoria optimizada en $1.86E+06$ unidades/KWH, obteniendo un aumento en la productividad del 85.65%.

3.4 TRAYECTORIA 3: FIGURA LÍNEAS RECTAS

La trayectoria Figura Líneas Rectas (Fig. 66), consta de 405 puntos y fue obtenida a partir del espacio cartesiano mediante un algoritmo de extracción de líneas 3D identificadas en un sólido modelado en Autodesk Inventor 2013® en conjunto con una orientación deseada validada por medio un análisis de la cinemática inversa de la trayectoria.

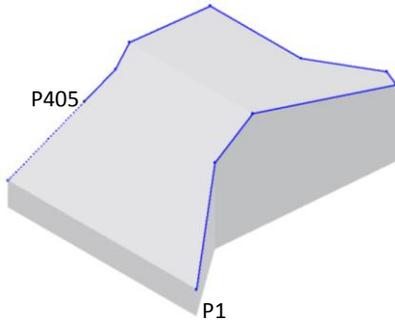


Figura 66 Trayectoria Figura Líneas Rectas

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Figura Líneas Rectas y se compara con la trayectoria sin optimizar, en el robot PUMA 560 con todos los parámetros dinámicos.

En la Fig. 67 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico a medida que el algoritmo optimiza la trayectoria Figura Líneas Rectas en cada iteración, realizada por el robot PUMA 560 con todos los parámetros dinámicos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 32.

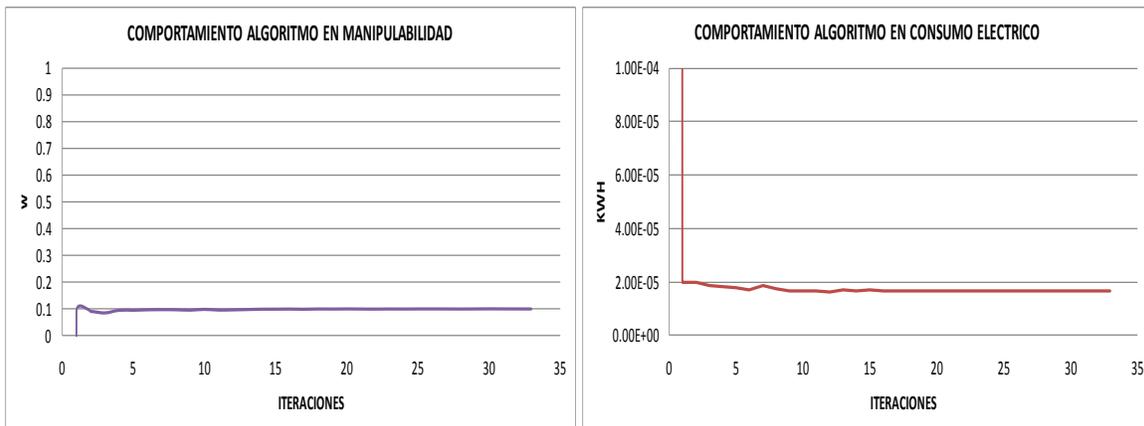


Figura 67 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 68 se identifican los recorridos que deben hacer las articulaciones del robot PUMA 560 con parámetros dinámicos y solo con parámetros cinemáticos, para seguir la trayectoria Figura Líneas Rectas.

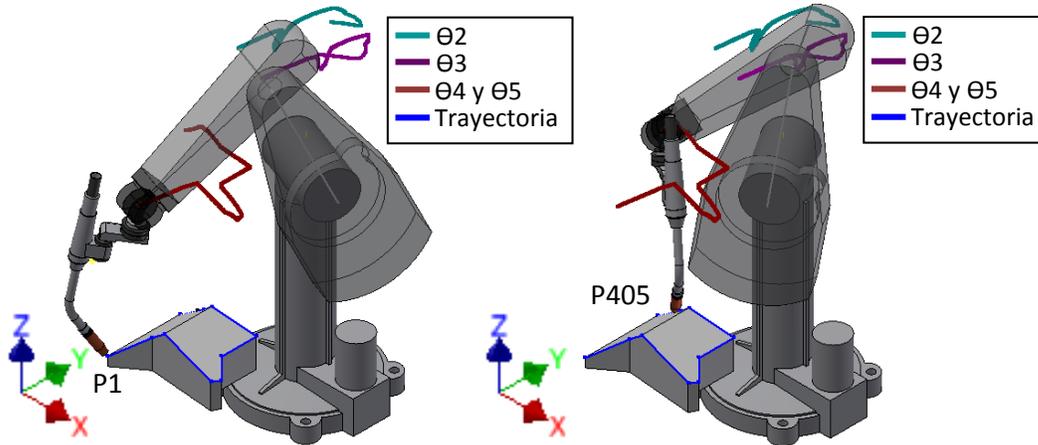


Figura 68 Trayectoria Figura Líneas Rectas en Robot PUMA 560

En la Fig. 69 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 con todos los parámetros dinámicos para seguir la trayectoria optimizada Figura Líneas Rectas. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.47589446 m, Angulo R: 0.67724003 rad y Angulo P: 0.37105104 rad.

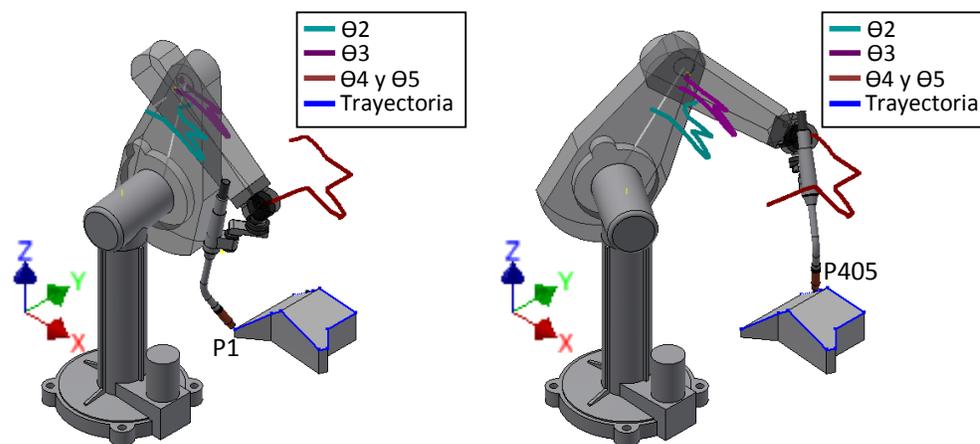


Figura 69 Trayectoria Figura Líneas Rectas en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos

En la Fig. 70 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Figura Líneas Rectas realizada por el robot PUMA 560 con todos los parámetros dinámicos.

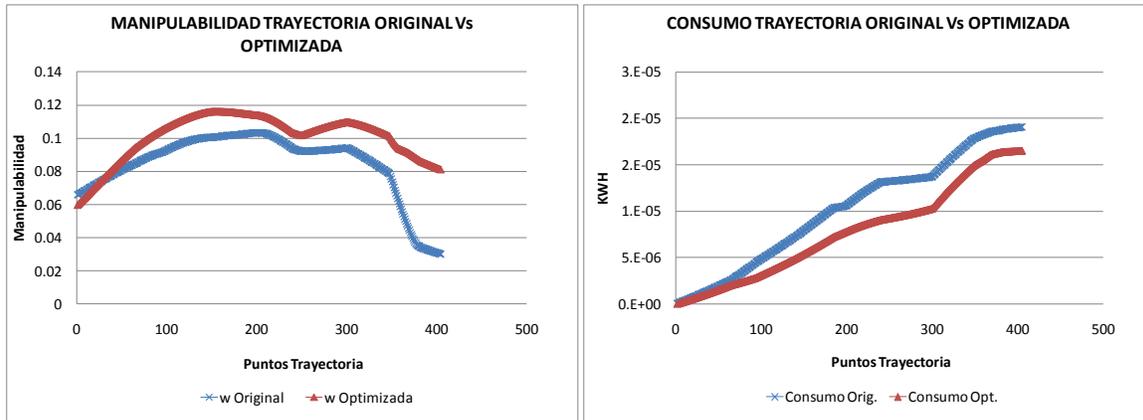


Figura 70 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot PUMA 560 con parámetros dinámicos

De la Fig. 70 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada hasta el punto 26 es menor que en la trayectoria original, pero luego se vuelve mayor y tiende a ser constante, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 47 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Figura Líneas Rectas original y optimizada del robot PUMA 560 con parámetros dinámicos, evidenciándose que el algoritmo de optimización en la iteración 32 logra optimizaciones del 13.25% en el consumo eléctrico, el 16.22% en la manipulabilidad, incluso del 5.74% en el tiempo de la trayectoria.

TRAYECTORIA FIGURA LÍNEAS RECTAS EN PUMA 560 CON PARAMETROS DINAMICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.73630183	1.90796E-05	0.08462562
Trayectoria Optimizada	32	1.63670240	1.65507E-05	0.10100740
% Optimización		5.74%	13.25%	16.22%

Tabla 47 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot PUMA 560 con parámetros dinámicos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 47 se puede identificar la productividad para la trayectoria original en 5.24E+04 unidades/KWH, y para la trayectoria optimizada en 6.04E+04 unidades/KWH, obteniendo un aumento en la productividad del 13.25%.

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Figura Líneas Rectas y se compara con la trayectoria sin optimizar, en el robot PUMA 560 solo con parámetros cinemáticos.

En la Fig. 71 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Figura Líneas Rectas en cada iteración, realizada por el robot PUMA 560 solo con parámetros cinemáticos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 27.

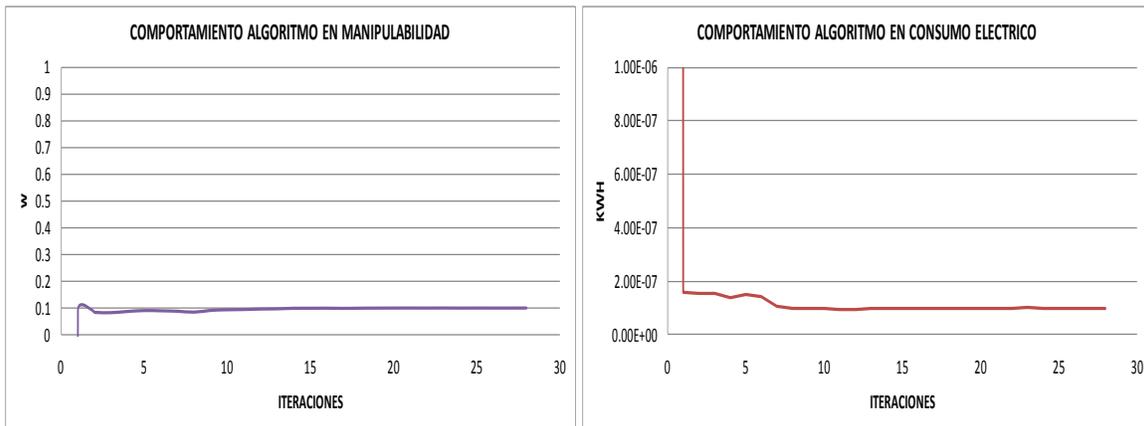


Figura 71 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 72 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 solo con parámetros cinemáticos para seguir la trayectoria optimizada Figura Líneas Rectas. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.86220728 m, Angulo R: 0.24711121 rad y Angulo P: -1.92829793 rad.

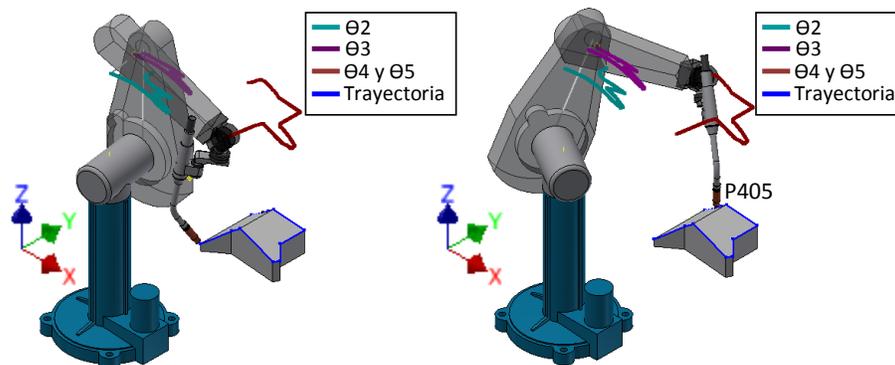


Figura 72 Trayectoria Figura Líneas Rectas Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 73 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Figura Líneas Rectas realizada por el robot PUMA 560 solo con parámetros cinemáticos.

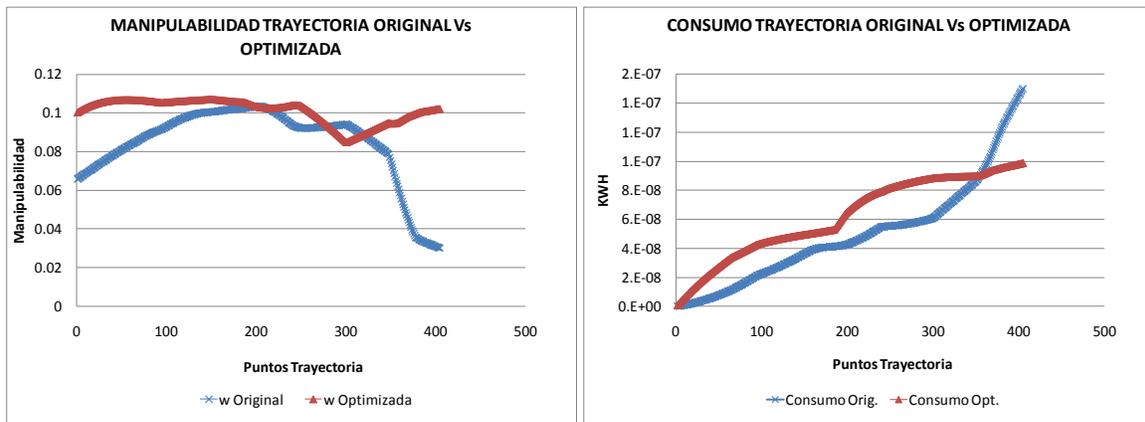


Figura 73 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 73 se aprecia que la manipulabilidad de la trayectoria optimizada, a pesar de ser menor que la obtenida con la trayectoria original en el tramo comprendido entre el punto 281 al 320, en el resto del recorrido es mayor y tiende a ser constante, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico de la trayectoria original, a pesar de ser menor hasta el punto 457 del recorrido frente al consumo de la trayectoria optimizada, se dispara abruptamente por lo cual termina siendo menor el consumo final total de la trayectoria optimizada

En la tabla 48 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Figura Líneas Rectas original y optimizada del robot PUMA 560 con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 27 logra optimizaciones del 34.16% en el consumo eléctrico, el 16.43% en la manipulabilidad, incluso del 4.71% en el tiempo de la trayectoria.

TRAYECTORIA FIGURA LÍNEAS RECTAS EN PUMA 560 CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	1.73630183	1.50036E-07	0.08462562
Trayectoria Optimizada	27	1.65452952	9.87889E-08	0.10126342
% Optimización		4.71%	34.16%	16.43%

Tabla 48 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 48 se puede identificar la productividad para la trayectoria original en $6.67E+06$ unidades/KWH, y para la trayectoria optimizada en $1.01E+07$ unidades/KWH, obteniendo un aumento en la productividad del 34.16%.

Con los datos del espacio de trabajo simplificado que generaron esta trayectoria, se procede a evaluar su comportamiento en el robot PUMA 560 con todos los parámetros dinámicos, la cual permite identificar que tan eficiente son estos datos de ubicación de la trayectoria en el espacio de trabajo en la optimización de los parámetros de consumo eléctrico y manipulabilidad. Estos resultados se muestran en la siguiente tabla:

TRAYECTORIA FIGURA LÍNEAS RECTAS EN PUMA 560 CON PARAMETROS DINAMICOS			
ROBOT	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original Parámetros dinámicos	1.73630183	1.90796E-05	0.08462562
Trayectoria Evaluada	1.65452952	1.86262E-05	0.10126342
% Optimización	4.71%	2.38%	16.43%

Tabla 49 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos

De la tabla 49 se evidencia que la trayectoria obtenida con el algoritmo de optimización en el robot PUMA 560 con solo parámetros cinemáticos, genera optimizaciones en el robot con todos los parámetros dinámicos aceptables: 2.38% en el consumo eléctrico y 16.43% en la manipulabilidad. El dato de la manipulabilidad obtenida es similar al aplicar el algoritmo para optimizar esta trayectoria en el robot PUMA 560 con todos los parámetros dinámicos, sin embargo el dato del consumo es 8.54% menor(tabla 47).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Figura Líneas Rectas y se compara con la trayectoria sin optimizar, en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

En la Fig. 74 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Figura Líneas Rectas en cada iteración, realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos En esta se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 28.

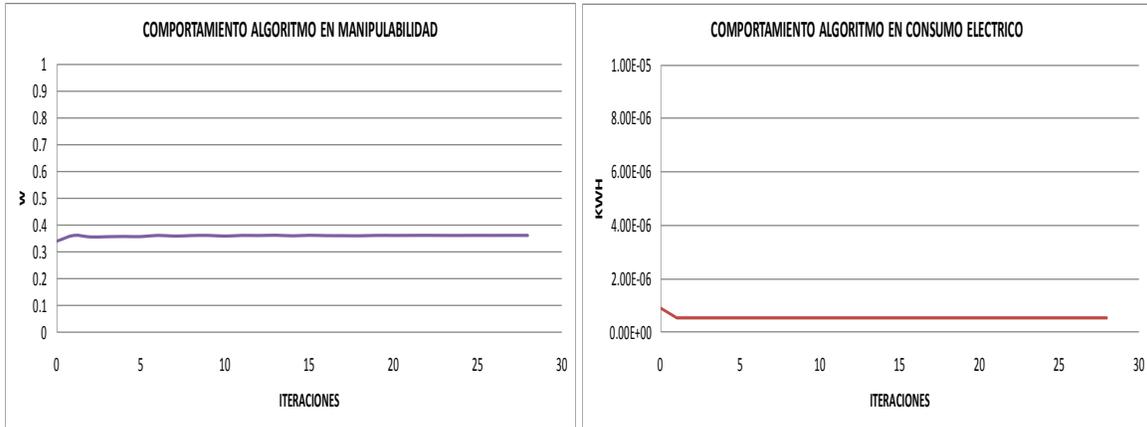


Figura 74 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Figura Líneas Rectas Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 75 se identifican los recorridos que deben hacer las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria Figura Líneas Rectas.

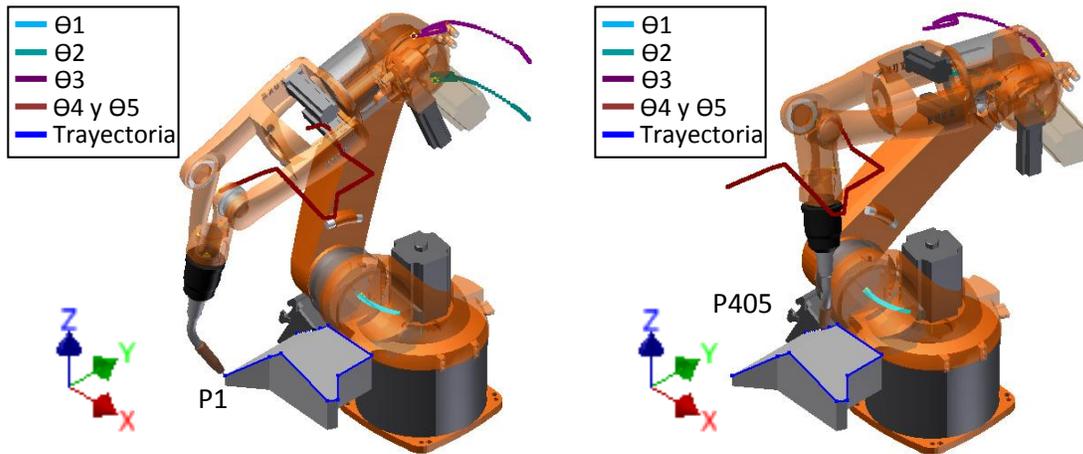


Figura 75 Trayectoria Figura Líneas Rectas en Robot KUKA KR5 HW

En la Fig. 76 se grafican los recorridos que deben realizar las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria optimizada Figura Líneas Rectas. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.92438664 m, Angulo R: 0.20065383 rad y Angulo P: 1.52360171 rad.

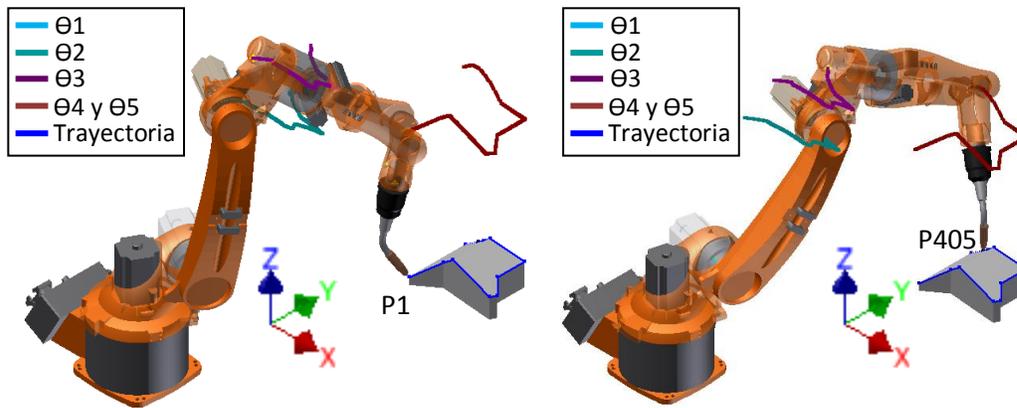


Figura 76 Trayectoria Figura Líneas Rectas Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 77 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Figura Líneas Rectas realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

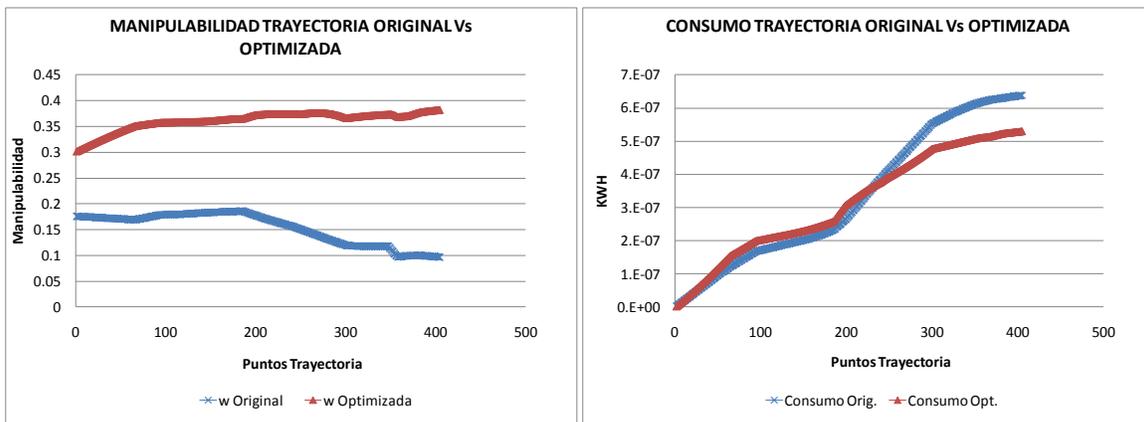


Figura 77 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Figura Líneas Rectas original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

De la Fig. 77 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la trayectoria original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico de la trayectoria original, a pesar de ser menor hasta el punto 233 del recorrido frente al consumo de la trayectoria optimizada, se dispara abruptamente por lo cual termina siendo menor el consumo final total de la trayectoria optimizada

En la tabla 50 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Figura Líneas Rectas original y optimizada del robot KUKA KR5 HW ARC con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 28 logra optimizaciones del 16.87% en el consumo eléctrico, el 57.70% en la manipulabilidad, pero el tiempo de la trayectoria aumenta en un 21.04%, este valor aumenta debido a que este no era un criterio a optimizar por el algoritmo.

TRAYECTORIA FIGURA LÍNEAS RECTAS EN KUKA KR5 HW ARC CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.67447332	6.37968E-07	0.15283169
Trayectoria Optimizada	28	2.02683357	5.30336E-07	0.36129511
% Optimización		-21.04%	16.87%	57.70%

Tabla 50 Comparación de parámetros optimizados en trayectoria Figura Líneas Rectas original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 50 se puede identificar la productividad para la trayectoria original en $1.57E+06$ unidades/KWH, y para la trayectoria optimizada en $1.89E+06$ unidades/KWH, obteniendo un aumento en la productividad del 16.87%.

3.5 TRAYECTORIA 4: HÉLICE UN PASO

Consta de 400 puntos y fue obtenida a partir del espacio cartesiano mediante un algoritmo de extracción de líneas 3D identificadas en un sólido modelado en Autodesk Inventor 2013® en conjunto con una orientación deseada validada por medio un análisis de la cinemática inversa de la trayectoria (Fig. 78).

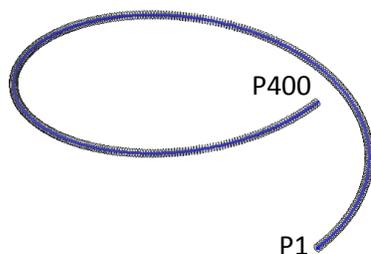


Figura 78 Trayectoria Hélice un Paso

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Hélice un Paso y se compara con la trayectoria sin optimizar, en el robot PUMA 560 con todos los parámetros dinámicos.

En la Fig. 79 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico a medida que el algoritmo optimiza la trayectoria Hélice un Paso en cada iteración, realizada por el robot PUMA 560 con todos los parámetros dinámicos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 41.

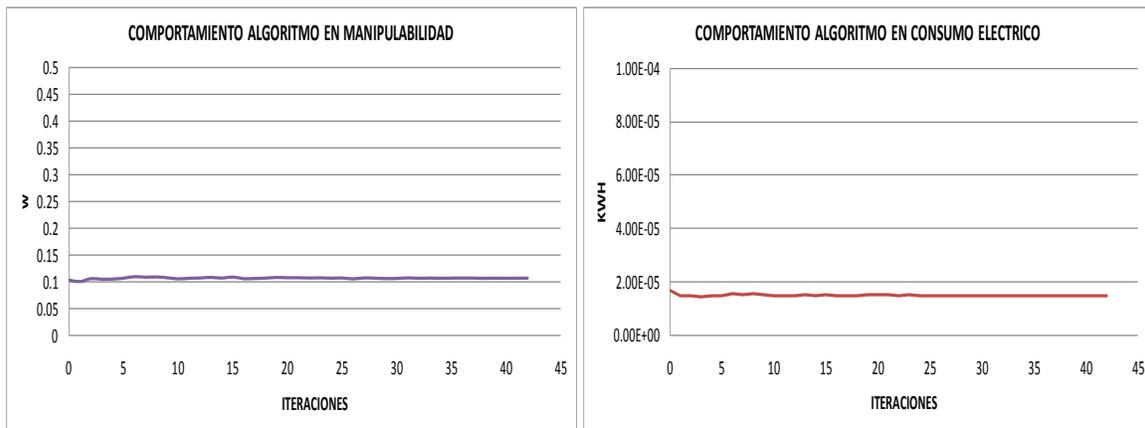


Figura 79 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 80 se identifican los recorridos que deben hacer las articulaciones del robot PUMA 560 con parámetros dinámicos y solo con parámetros cinemáticos, para seguir la trayectoria Hélice un Paso.

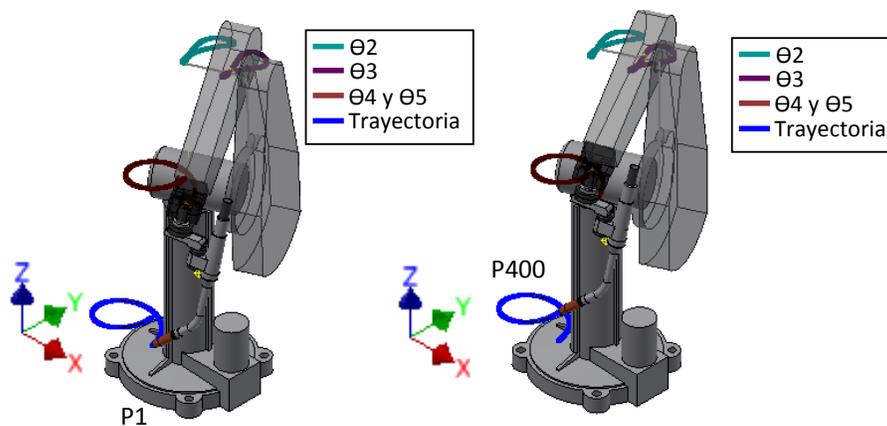


Figura 80 Trayectoria Hélice un Paso en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos

En la Fig. 81 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 con todos los parámetros dinámicos para seguir la trayectoria optimizada Hélice un Paso. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.63654636 m, Angulo R: 0.64467012 rad y Angulo P: 1.689265 rad.

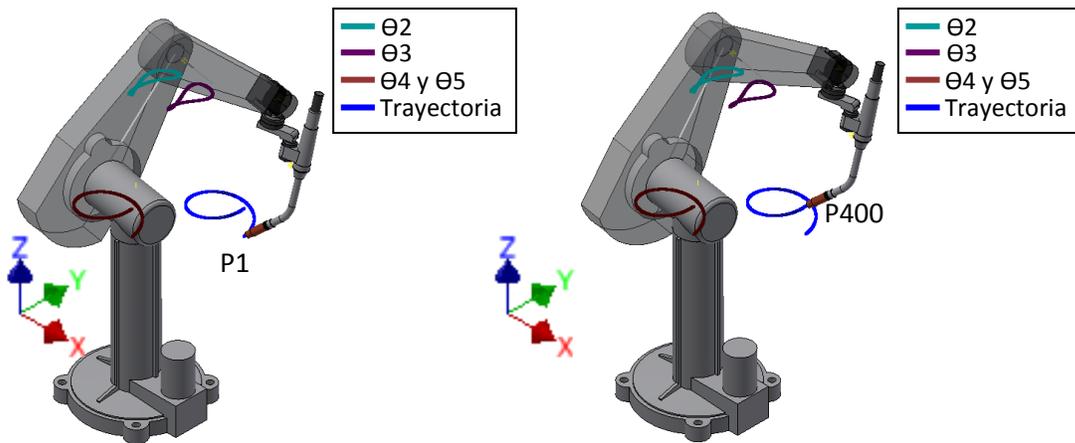


Figura 81 Trayectoria Hélice un Paso Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 82 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Hélice un Paso realizada por el robot PUMA 560 con todos los parámetros dinámicos.

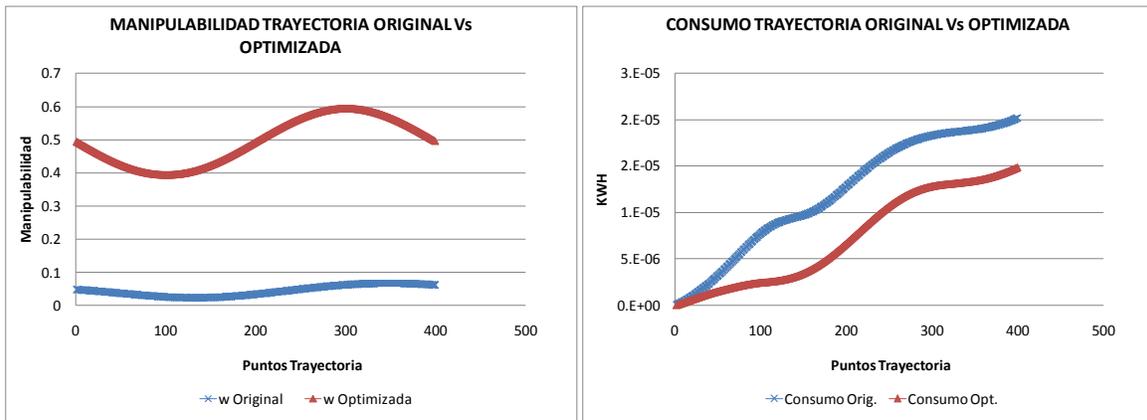


Figura 82 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot PUMA 560 con parámetros dinámicos

De la Fig. 82 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la trayectoria original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de

la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 51 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Hélice un Paso original y optimizada del robot PUMA 560 con parámetros dinámicos, evidenciándose que el algoritmo de optimización en la iteración 41 logra optimizaciones del 26.45% en el consumo eléctrico, el 58.30% en la manipulabilidad, pero no presenta optimización alguna en el tiempo de la trayectoria (0%) debido a que este no era un criterio a optimizar por el algoritmo.

TRAYECTORIA HÉLICE UN PASO EN PUMA 560 CON PARAMETROS DINAMICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	1.26927302	2.01737E-05	0.04475938
Trayectoria Optimizada	41	1.26925917	1.48384E-05	0.10734889
% Optimización		0.00%	26.45%	58.30%

Tabla 51 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot PUMA 560 con parámetros dinámicos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 51 se puede identificar la productividad para la trayectoria original en $4.96E+04$ unidades/KWH, y para la trayectoria optimizada en $6.74E+04$ unidades/KWH, obteniendo un aumento en la productividad del 26.45%.

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Hélice un Paso y se compara con la trayectoria sin optimizar, en el robot PUMA 560 solo con parámetros cinemáticos.

En la Fig. 83 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Hélice un Paso en cada iteración, realizada por el robot PUMA 560 solo con parámetros cinemáticos En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 22.

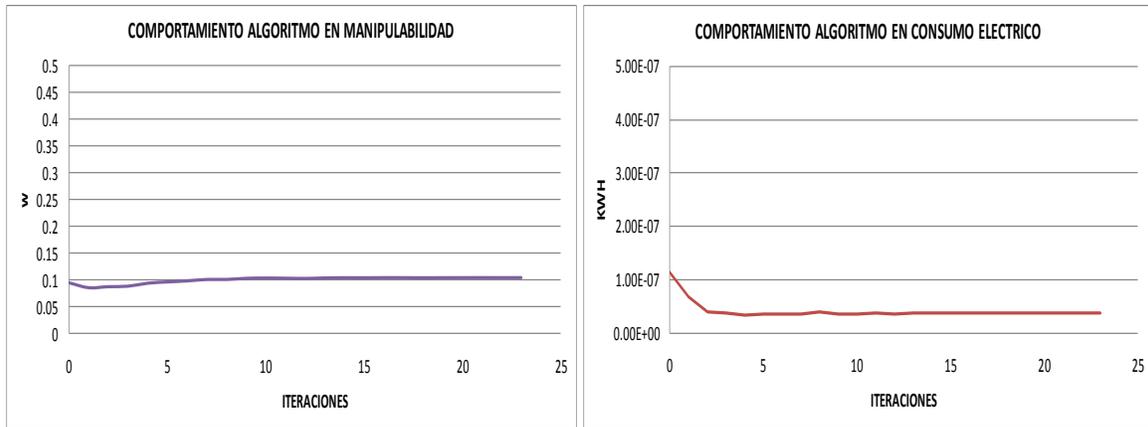


Figura 83 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 84 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 solo con parámetros cinemáticos para seguir la trayectoria optimizada Hélice un Paso. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.7793329 m, Angulo R: 0.32708995 rad y Angulo P: -0.08509418 rad.

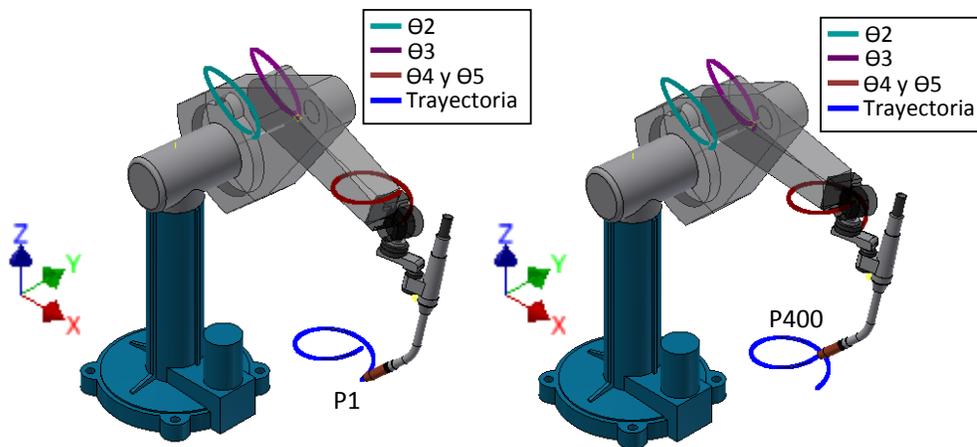


Figura 84 Trayectoria Hélice un Paso Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 85 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Hélice un Paso realizada por el robot PUMA 560 solo con parámetros cinemáticos.

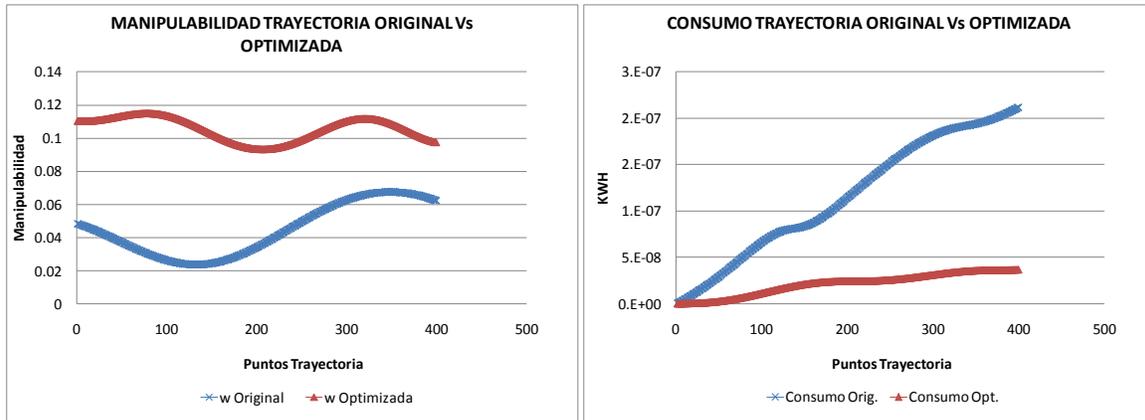


Figura 85 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot PUMA 560 solo con parámetros cinemáticos

De la Fig. 85 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor y tiende a ser constante a lo largo del recorrido, comparándola con la trayectoria original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 52 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Hélice un Paso original y optimizada del robot PUMA 560 con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 22 logra optimizaciones del 82.46% en el consumo eléctrico, el 57.63% en la manipulabilidad, pero no presenta optimización alguna en el tiempo de la trayectoria (0%) debido a que este no era un criterio a optimizar por el algoritmo.

TRAYECTORIA HELICE UN PASO EN PUMA 560 CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	1.26927302	2.11748E-07	0.04475938
Trayectoria Optimizada	22	1.26925917	3.71365E-08	0.10564592
% Optimización		0.00%	82.46%	57.63%

Tabla 52 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 52 se puede identificar la productividad para la trayectoria original en 2.69E+07 unidades/KWH, y para

la trayectoria optimizada en $4.72E+06$ unidades/KWH, obteniendo un aumento en la productividad del 82.46%.

Con los datos del espacio de trabajo simplificado que generaron esta trayectoria, se procede a evaluar su comportamiento en el robot PUMA 560 con todos los parámetros dinámicos, la cual permite identificar que tan eficiente son estos datos de ubicación de la trayectoria en el espacio de trabajo en la optimización de los parámetros de consumo eléctrico y manipulabilidad. Estos resultados se muestran en la siguiente tabla:

TRAYECTORIA HELICE UN PASO EN PUMA 560 CON PARAMETROS DINAMICOS			
ROBOT	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original Parámetros dinámicos	1.26927302	2.01737E-05	0.04475938
Trayectoria Evaluada	1.26925917	1.83227E-05	0.10564592
% Optimización	0.00%	9.18%	57.63%

Tabla 53 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos

De la tabla 53 se evidencia que la trayectoria obtenida con el algoritmo de optimización en el robot PUMA 560 con solo parámetros cinemáticos, genera optimizaciones en el robot con todos los parámetros dinámicos aceptables: 9.18% en el consumo eléctrico y 57.63% en la manipulabilidad. El dato de la manipulabilidad obtenido es similar al aplicar el algoritmo para optimizar esta trayectoria en el robot PUMA 560 con todos los parámetros dinámicos, sin embargo el dato del consumo es 17.27% menor (tabla 51).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Hélice un Paso y se compara con la trayectoria sin optimizar, en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

En la Fig. 86 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Hélice un Paso en cada iteración, realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos. En esta se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 26.

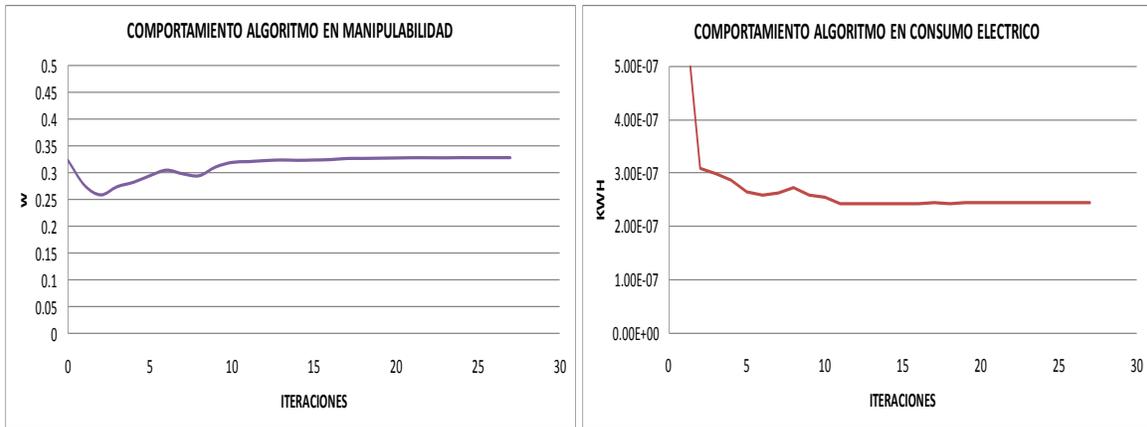


Figura 86 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Hélice un Paso Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 87 se identifican los recorridos que deben hacer las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria Hélice un Paso

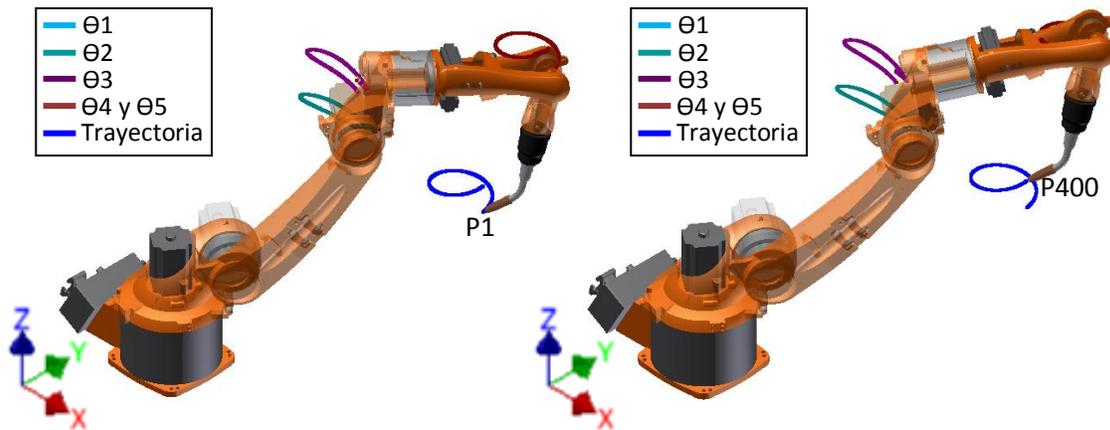


Figura 87 Trayectoria Hélice un Paso sin optimizar en Robot KUKA KR5 HW

En la Fig. 88 se grafican los recorridos que deben realizar las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos para seguir la trayectoria optimizada Hélice un Paso. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 1.15386433 m, Angulo R: -0.00314029 rad y Angulo P: -0.97583093 rad.

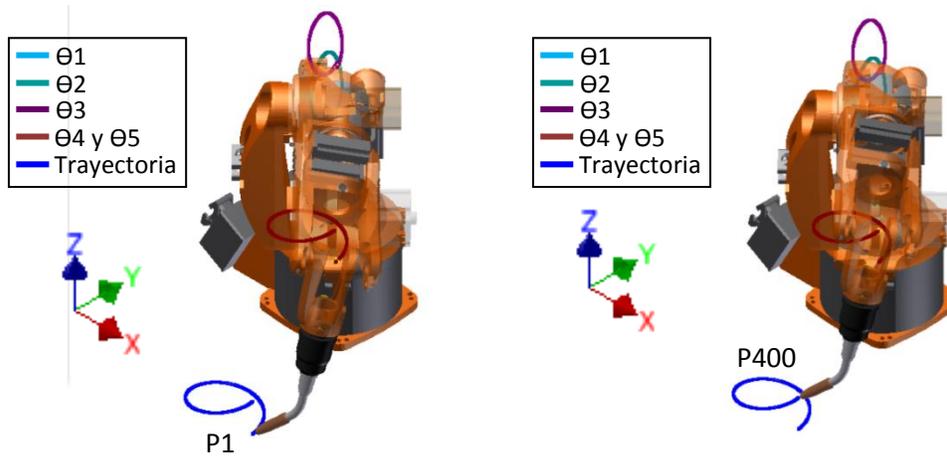


Figura 88 Trayectoria Hélice un Paso Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 89 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Hélice un Paso realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

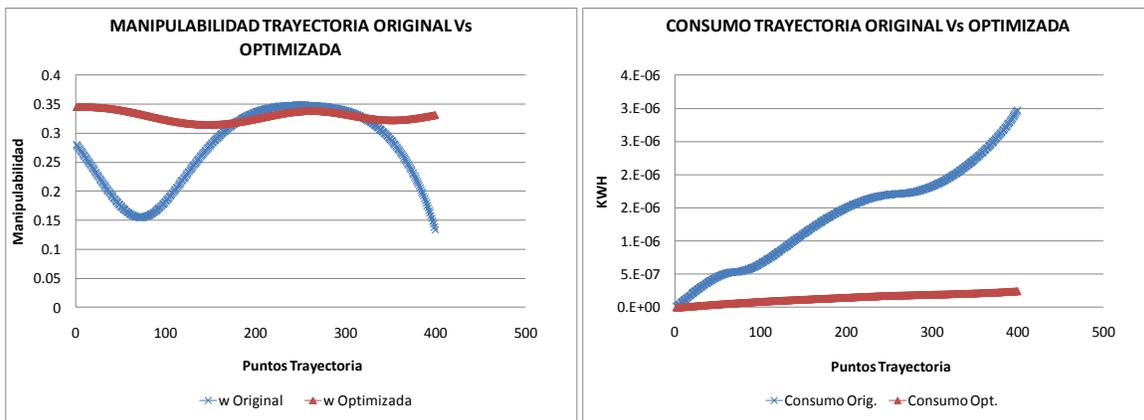


Figura 89 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Hélice un Paso original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

De la Fig. 89 se aprecia que la manipulabilidad de la trayectoria optimizada en el tramo del punto 179 al punto 319 es ligeramente menor que la manipulabilidad de la trayectoria original, pero en promedio la manipulabilidad de la trayectoria optimizada es mayor y tiene a ser constante. Esta situación es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 54 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Hélice un Paso original y optimizada del robot KUKA KR5 HW ARC con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 26 logra optimizaciones del 91.75% en el consumo eléctrico, el 17.71% en la manipulabilidad, pero no presenta optimización alguna en el tiempo de la trayectoria (0%) debido a que este no era un criterio a optimizar por el algoritmo.

TRAYECTORIA HÉLICE UN PASO EN KUKA KR5 HW ARC CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	1.26925917	2.97369E-06	0.27089670
Trayectoria Optimizada	26	1.26925917	2.45245E-07	0.32919802
% Optimización		0.00%	91.75%	17.71%

Tabla 54 Comparación de parámetros optimizados en trayectoria Hélice un Paso original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 54 se puede identificar la productividad para la trayectoria original en $3.36E+05$ unidades/KWH, y para la trayectoria optimizada en $4.08E+06$ unidades/KWH, obteniendo un aumento en la productividad del 91.75%.

3.6 TRAYECTORIA 5: GABINETE

Consta de 694 puntos y fue obtenida a partir del espacio cartesiano mediante un algoritmo de extracción de líneas 3D identificadas en un sólido modelado en Autodesk Inventor 2013® en conjunto con una orientación deseada validada por medio un análisis de la cinemática inversa de la trayectoria. Esta trayectoria es el típico ejemplo del trabajo de un robot soldador en una línea de soldadura (Fig. 90).

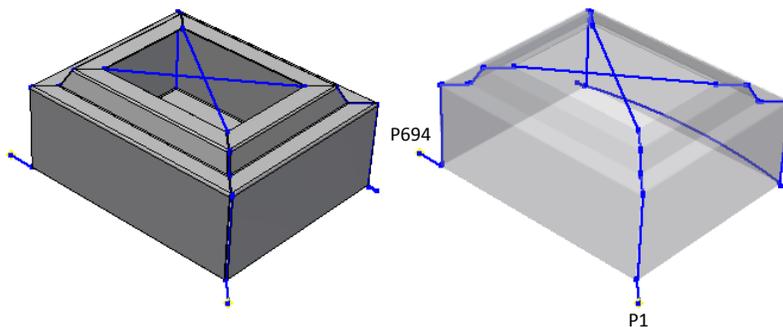


Figura 90 Trayectoria Gabinete

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Gabinete y se compara con la trayectoria sin optimizar, en el robot PUMA 560 con todos los parámetros dinámicos.

En la Fig. 91 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico a medida que el algoritmo optimiza la trayectoria Gabinete en cada iteración, realizada por el robot PUMA 560 con todos los parámetros dinámicos. En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 44.

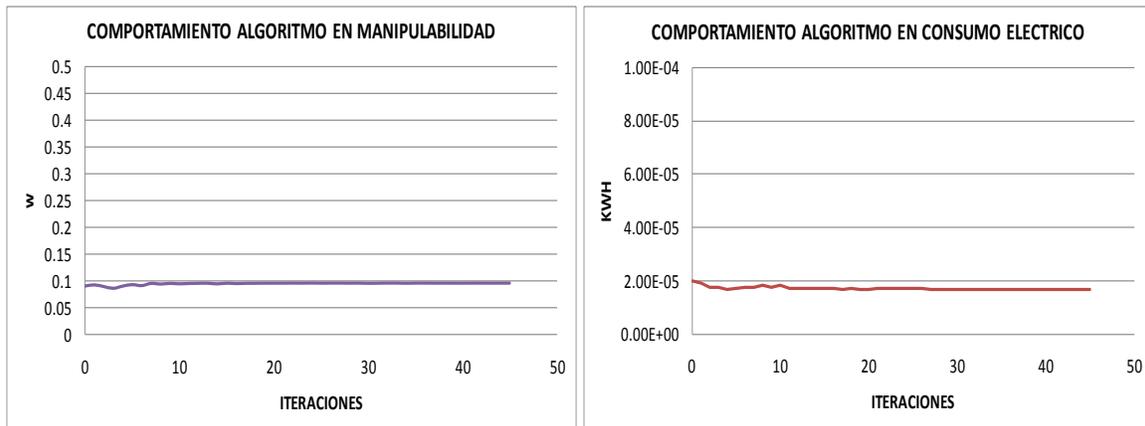


Figura 91 Manipulabilidad Vs iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 92 se identifican los recorridos que deben hacer las articulaciones del robot PUMA 560 con parámetros dinámicos y solo con parámetros cinemáticos, para seguir la trayectoria Gabinete.

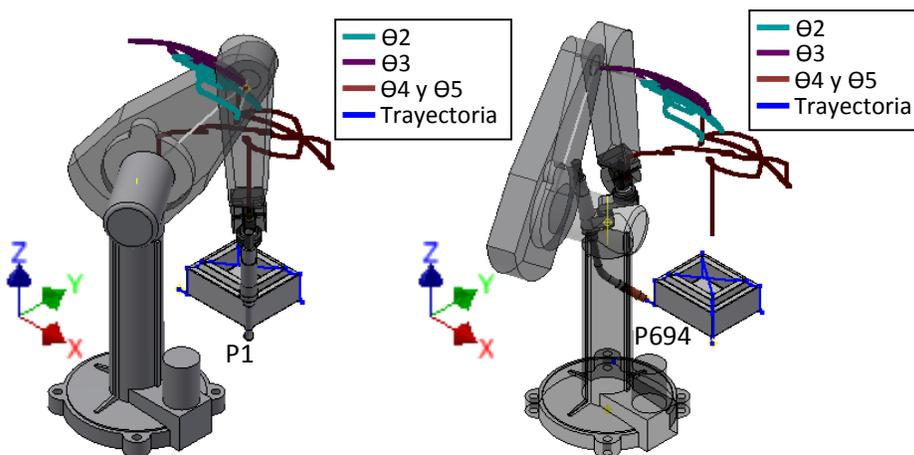


Figura 92 Trayectoria Gabinete en Robot PUMA 560 con parametros dinámicos y solo con parámetros cinemáticos

En la Fig. 93 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 con todos los parámetros dinámicos para seguir la trayectoria optimizada Gabinete. . Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.5497875 m, Angulo R: 0.56249842 rad y Angulo P: 1.34549279 rad.

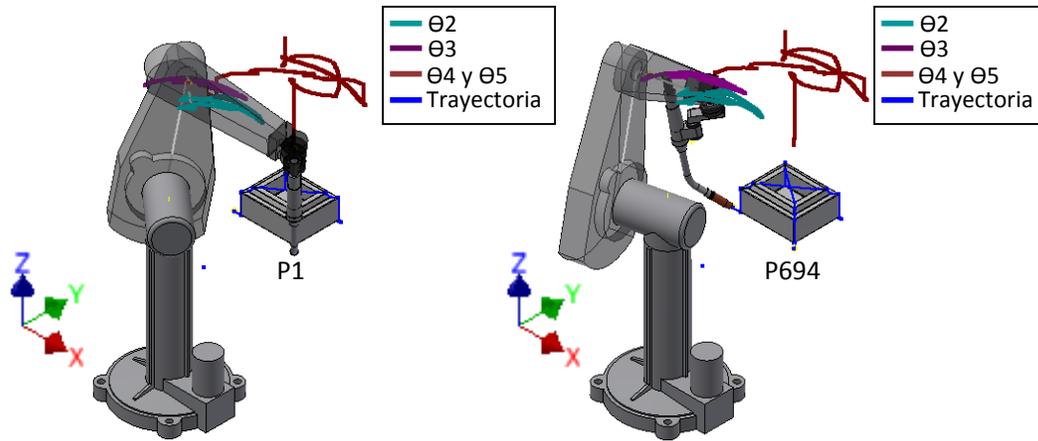


Figura 93 Trayectoria Gabinete Optimizada en Robot PUMA 560 con parámetros dinámicos

En la Fig. 94 se presentan las gráficas del comportamiento de la manipulabilidad y del consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Gabinete realizada por el robot PUMA 560 con todos los parámetros dinámicos.

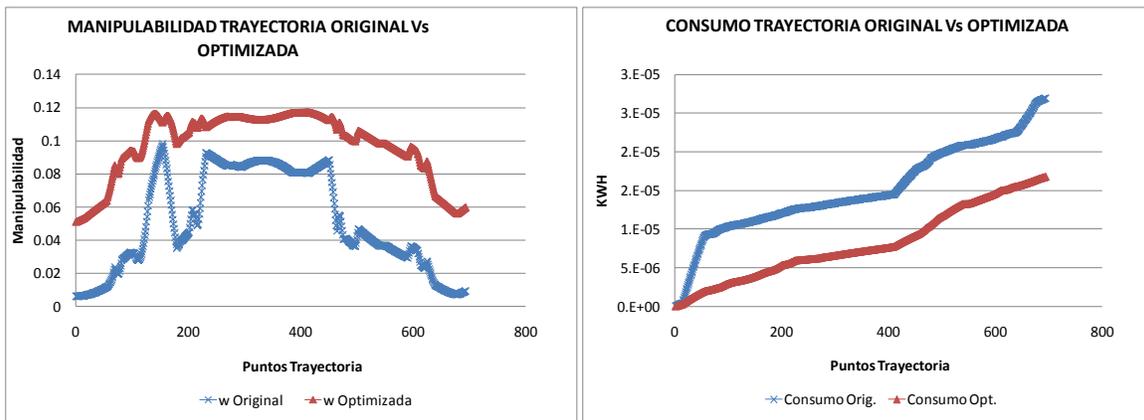


Figura 94 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete original vs optimizada en Robot PUMA 560 con parámetros dinámicos

De la Fig. 94 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor a lo largo del recorrido, comparándola con la trayectoria original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria

optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 55 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Gabinete original y optimizada del robot PUMA 560 con parámetros dinámicos, evidenciándose que el algoritmo de optimización en la iteración 44 logra optimizaciones del 37.76% en el consumo eléctrico, el 47.51% en la manipulabilidad, incluso del 1.30% en el tiempo de la trayectoria.

TRAYECTORIA GABINETE EN PUMA 560 CON PARAMETROS DINAMICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	3.76784326	2.69323E-05	0.05115588
Trayectoria Optimizada	44	3.71877864	1.67633E-05	0.09746307
% Optimización		1.30%	37.76%	47.51%

Tabla 55 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot PUMA 560 con parámetros dinámicos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 55 se puede identificar la productividad para la trayectoria original en $3.71E+04$ unidades/KWH, y para la trayectoria optimizada en $5.97E+04$ unidades/KWH, obteniendo un aumento en la productividad 37.76%.

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Gabinete y se compara con la trayectoria sin optimizar, en el robot PUMA 560 solo con parámetros cinemáticos.

En la Fig. 95 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Gabinete en cada iteración, realizada por el robot PUMA 560 solo con parámetros cinemáticos En estas se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 22.

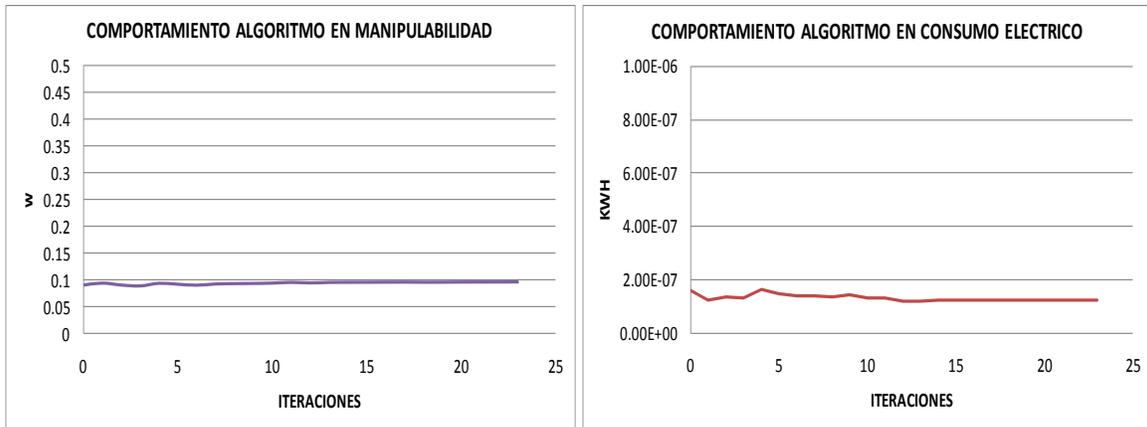


Figura 95 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 96 se grafican los recorridos que deben realizar las articulaciones del robot PUMA 560 solo con parámetros cinemáticos para seguir la trayectoria optimizada Gabinete. Para obtener esta trayectoria los datos del espacio de trabajo simplificado que usa el algoritmo para definir el punto inicial son: Radio = 0.59055267 m, Angulo R0.57577478 rad y Angulo P: 1.26274614 rad.

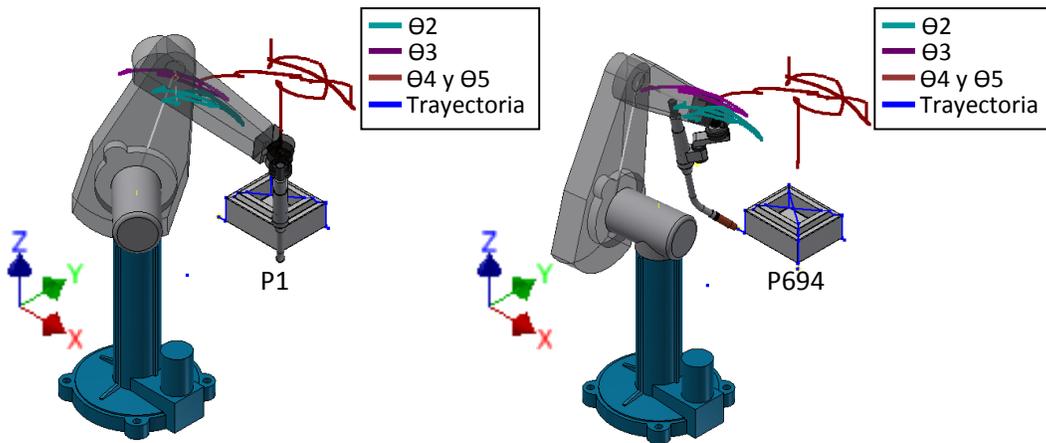


Figura 96 Trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos

En la Fig. 97 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Gabinete realizada por el robot PUMA 560 solo con parámetros cinemáticos.

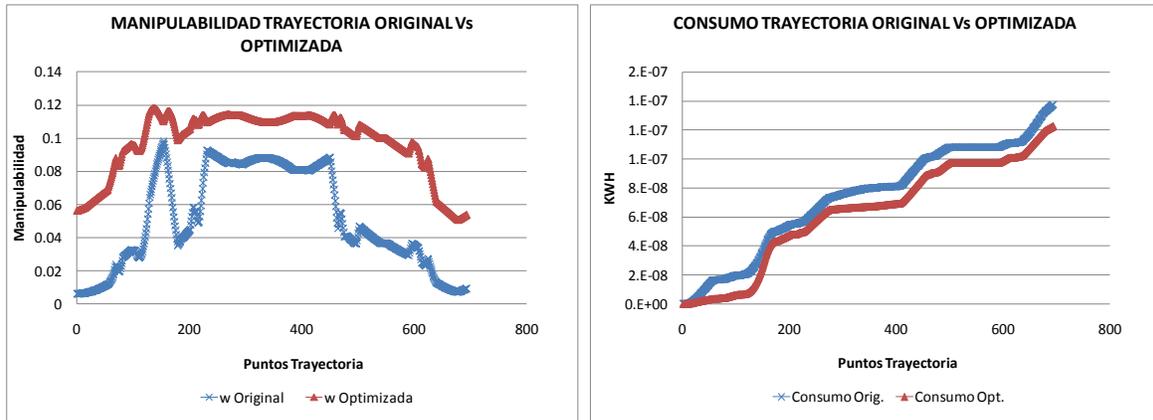


Figura 97 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete Optimizada en Robot PUMA 560 solo parámetros cinemáticos

De la Fig. 97 se aprecia que la manipulabilidad obtenida con la trayectoria optimizada es mayor a lo largo del recorrido, comparándola con la trayectoria original, situación que es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que el consumo eléctrico a lo largo del recorrido de la trayectoria optimizada siempre es menor que el consumo de la trayectoria original, al igual que el consumo final total.

En la tabla 56 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Gabinete original y optimizada del robot PUMA 560 con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 22 logra optimizaciones del 10.92% en el consumo eléctrico, el 47.51% en la manipulabilidad, pero no presenta optimización significativa en el tiempo de la trayectoria (0.02%) debido a que este no era un criterio a optimizar por el algoritmo.

TRAYECTORIA GABINETE EN PUMA 560 CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	----	3.76784326	1.37541E-07	0.05115588
Trayectoria Optimizada	22	3.76702613	1.22526E-07	0.09745148
% Optimización		0.02%	10.92%	47.51%

Tabla 56 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot PUMA 560 solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 56 se puede identificar la productividad para la trayectoria original en 7.27E+06 unidades/KWH, y para

la trayectoria optimizada en $8.16E+06$ unidades/KWH, obteniendo un aumento en la productividad del 10.92%.

Con los datos del espacio de trabajo simplificado que generaron esta trayectoria, se procede a evaluar su comportamiento en el robot PUMA 560 con todos los parámetros dinámicos, la cual permite identificar que tan eficiente son estos datos de ubicación de la trayectoria en el espacio de trabajo en la optimización de los parámetros de consumo eléctrico y manipulabilidad. Estos resultados se muestran en la siguiente tabla:

TRAYECTORIA GABINETE EN PUMA 560 CON PARAMETROS DINAMICOS			
ROBOT	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original Parámetros dinámicos	3.76784326	2.69323E-05	0.05115588
Trayectoria Evaluada	3.76702613	1.74291E-05	0.09745148
% Optimización	0.02%	35.29%	47.51%

Tabla 57 Comparación de parámetros optimizados en trayectoria Gabinete original contra trayectoria evaluada en robot PUMA 560 con parámetros dinámicos

De la tabla 57 se evidencia que la trayectoria obtenida con el algoritmo de optimización en el robot PUMA 560 con solo parámetros cinemáticos, genera optimizaciones en el robot con todos los parámetros dinámicos muy satisfactorias: 35.29% en el consumo eléctrico y 47.51% en la manipulabilidad. Estos resultados son similares a los obtenidos al aplicar el algoritmo para optimizar esta trayectoria en el robot PUMA 560 con todos los parámetros dinámicos (tabla 55).

A continuación se presentan los resultados del comportamiento del algoritmo de optimización con los parámetros de calibración seleccionados, aplicado en la trayectoria Gabinete y se compara con la trayectoria sin optimizar, en el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

En la Fig. 98 se presentan las gráficas del comportamiento de la manipulabilidad y el consumo eléctrico a medida que el algoritmo optimiza la trayectoria Gabinete en cada iteración, realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos. En esta se aprecia que la manipulabilidad y el consumo se estabilizan en la iteración 42.

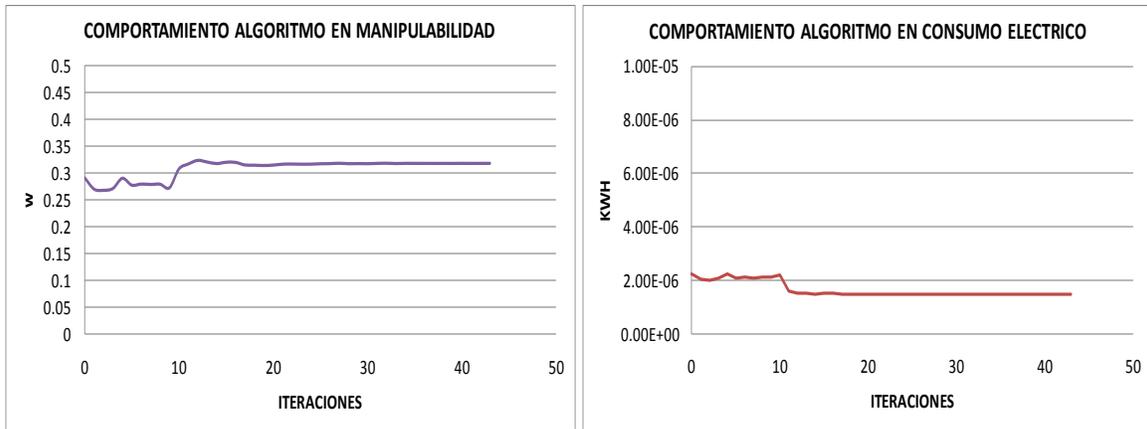


Figura 98 Manipulabilidad Vs Iteraciones y consumo eléctrico Vs iteraciones de trayectoria Gabinete Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 99 se identifican los recorridos que deben hacer las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos, para seguir la trayectoria Gabinete.

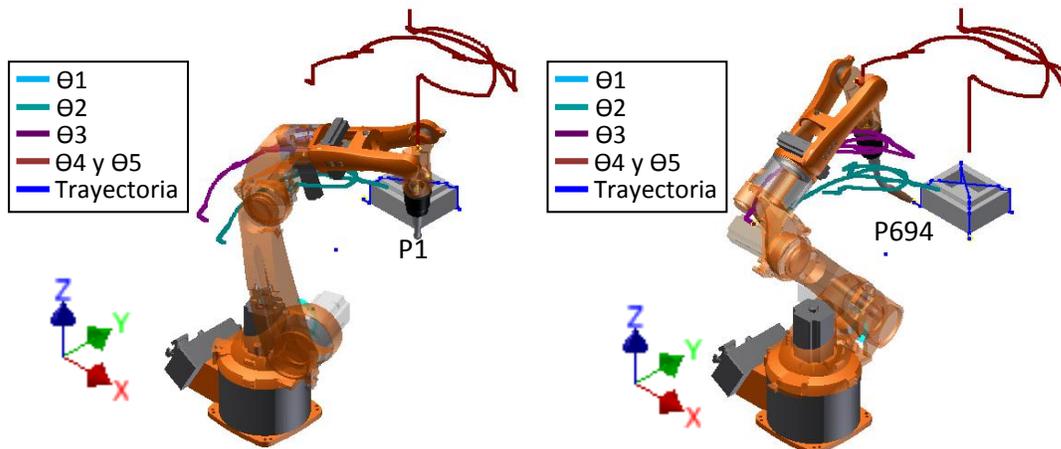


Figura 99 Trayectoria Gabinete sin optimizar en Robot KUKA KR5 HW

En la Fig. 100 se grafican los recorridos que deben realizar las articulaciones del robot KUKA KR5 HW ARC solo con parámetros cinemáticos para seguir la trayectoria optimizada Gabinete.

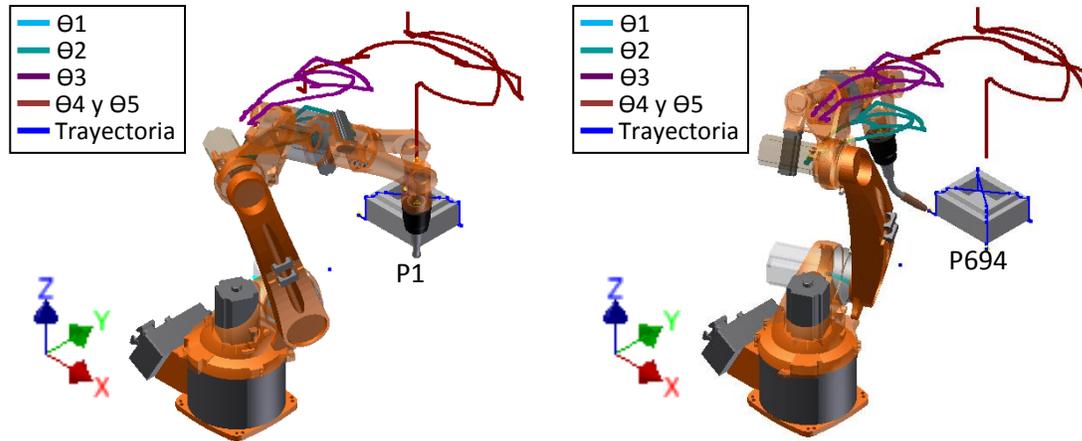


Figura 100 Trayectoria Gabinete Optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

En la Fig. 101 se presentan las gráficas del comportamiento de la manipulabilidad y consumo eléctrico en la trayectoria original, contra la trayectoria optimizada Gabinete realizada por el robot KUKA KR5 HW ARC solo con parámetros cinemáticos.

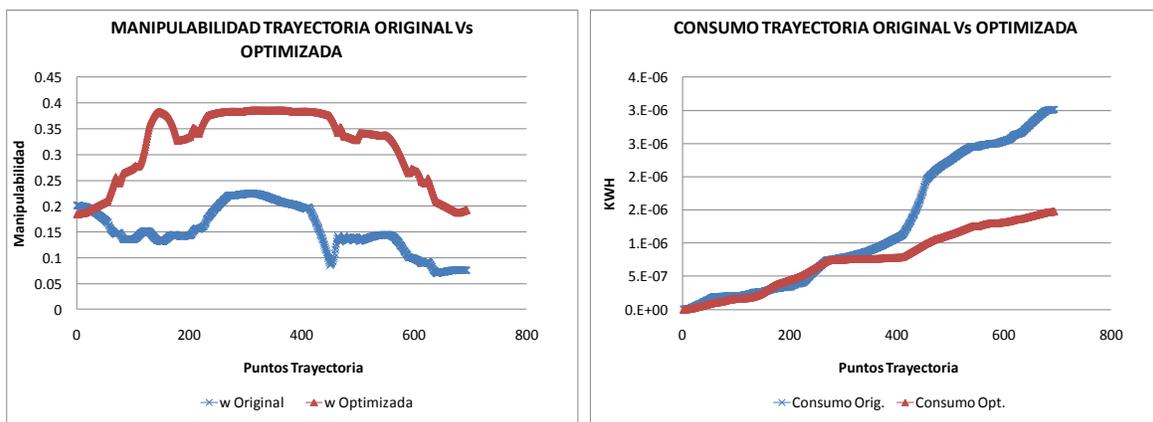


Figura 101 Comparación manipulabilidad y comparación consumo eléctrico de trayectoria Gabinete original vs optimizada en Robot KUKA KR5 HW ARC solo parámetros cinemáticos

De la Fig. 101 se aprecia que desde el inicio del recorrido hasta el punto 24, la manipulabilidad de la trayectoria original es mayor que la trayectoria optimizada. Sin embargo a partir de ese punto la manipulabilidad de la trayectoria optimizada comienza a aumentar conservándose siempre mayor que la manipulabilidad de la trayectoria original. Esta situación es favorable porque indica un aumento en el desempeño del manipulador en la ejecución de la trayectoria. Adicionalmente se aprecia que desde el inicio del recorrido hasta el punto 275, el comportamiento del consumo eléctrico de ambas trayectorias es muy similar. Sin embargo a partir de ese punto el consumo de eléctrico de la trayectoria optimizada empieza a ser menor conservándose así hasta el final del recorrido, obteniéndose, por consiguiente, un consumo final menor.

En la tabla 58 se muestran los valores de los criterios de consumo eléctrico y la manipulabilidad promedio de la trayectoria Gabinete original y optimizada del robot KUKA KR5 HW ARC con solo parámetros cinemáticos, evidenciándose que el algoritmo de optimización en la iteración 42 logra optimizaciones del 50.93% en el consumo eléctrico, el 51.63% en la manipulabilidad, incluso del 2.67% en el tiempo de la trayectoria.

TRAYECTORIA GABINETE EN KUKA KR5 HW ARC CON SOLO PARAMETROS CINEMATICOS				
ROBOT	Iteración	Tiempo Trayectoria (seg)	Consumo (KWH)	w Promedio
Trayectoria Original	-----	3.01145583	3.01627E-06	0.15397083
Trayectoria Optimizada	42	2.93103530	1.48002E-06	0.31829922
% Optimización		2.67%	50.93%	51.63%

Tabla 58 Comparación de parámetros optimizados en trayectoria Gabinete original contra optimizada en robot KUKA KR5 HW ARC solo con parámetros cinemáticos

Partiendo de la ecuación(79) de la productividad de energía y de los valores de la tabla 58 se puede identificar la productividad para la trayectoria original en $3.32E+05$ unidades/KWH, y para la trayectoria optimizada en $6.76E+05$ unidades/KWH, obteniendo un aumento en la productividad del 50.93%.

CONCLUSIONES Y RECOMENDACIONES

En la presente investigación se propone un método multi-objetivo basado en el Algoritmo Heurístico de Kalman, el cual integra simultáneamente la optimización de consumo eléctrico y manipulabilidad, obteniendo una estrategia de planeación de trayectorias para manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica, usados en procesos soldadura de arco en lámina el cual permite un aumento en la productividad respecto a las trayectorias programadas por un usuario.

La estrategia de optimización se basa en la búsqueda de la ubicación de una trayectoria base dentro de un espacio de trabajo esférico simplificado, la cual conlleva a que los criterios de manipulabilidad y de consumo eléctrico se optimicen, generando recorridos suaves a velocidades constantes en el efector final, gracias a que las articulaciones del robot estarán alejadas de posibles singularidades, maximizando su destreza y disminuyendo el esfuerzo soportado en éstas (índice de manipulabilidad tiende a 1), y cuidando, adicionalmente, que la cinemática y dinámica se encuentren dentro de los límites propios del manipulador.

El método fue validado mediante simulaciones del desplazamiento del manipulador a través de las trayectorias originales y optimizadas mediante un algoritmo implementado en la aplicación de Visual Basic perteneciente al software Autodesk Inventor 2013.

De acuerdo los datos obtenidos en las tablas ilustradas en la sección de resultados, se evidencia que el algoritmo heurístico de Kalman es viable para ser usado como base para desarrollar métodos de optimización multicriterios de trayectorias debido a las diversas ventajas que presenta, entre ellas, su facilidad de implementación, la demanda de baja memoria y requisitos de velocidad de procesador, procedimientos de búsqueda basados únicamente en los valores de la función objetivo, sin necesidad de suposiciones fuertes tales como linealidad, diferenciabilidad, convexidad, etc. Uno de los inconvenientes detectados con el algoritmo heurístico de Kalman es su convergencia en soluciones locales, pero esto se sorteó modificando la inicialización del algoritmo.

El enfoque de planeación de trayectoria propuesto permite ser usado en manipuladores antropomórficos de seis grados de libertad y muñeca esférica, de los cuales se conozca sus parámetros cinemáticos y dinámicos, para generar trayectorias optimizadas desde los criterios de manipulabilidad y energía eléctrica, conservando la orientación del efector final. Debido al planteamiento total de los modelos matemáticos de los parámetros cinemáticos, diferencial y dinámicos para manipuladores seriales antropomórficos de seis grados de libertad y muñeca esférica la función de costo del algoritmo puede ser fácilmente cambiada por otros dos criterios diferentes (dimensiones, velocidades y/o aceleraciones de los eslabones, torques eslabones o de los motores, relaciones de transmisión, tiempos de recorrido, entre otros). Gracias a la modularidad con la cual el enfoque fue desarrollado sería posible cambiar fácilmente el método de optimización HKA por otro método metaheurístico.

Una ventaja importante que posee el método es que a partir de una trayectoria planteada en forma de coordenadas cartesianas XYZ y orientaciones del efector en ángulos tipo Euler (precesión, nutación y rotación propia), buscará una solución óptima posible dentro del espacio de trabajo simplificado del manipulador respetando sus límites articulares y de velocidad, permitiendo con esto tener trayectorias de entrada en las cuales solo importe la orientación de la herramienta a lo largo del recorrido situación indispensable para el proceso de soldadura.

Como limitante del método está la identificación de los valores de todos los parámetros dinámicos del manipulador comercial necesarios para el cálculo correcto del consumo eléctrico, debido a que los fabricantes solo proporcionan los datos básicos del manipulador. Estos parámetros son propios del diseño del manipulador y sus componentes por lo cual identificarlos es una tarea para la cual se debe contar con las herramientas necesarias para desmantelar el manipulador y cuantificarlos. Sin embargo el método sería ideal para ser usado como herramienta de evaluación de parámetros cuando se esté diseñado totalmente un manipulador serial antropomórfico de seis grados de libertad y muñeca esférica.

Cabe anotar que el método puede ser aplicado en manipuladores antropomórficos de los cuales se desconozca sus parámetros dinámicos, con la condición que las trayectorias resultantes sean usadas sólo para realizar análisis cinemáticos, porque alguno de los parámetros dinámicos desconocidos del manipulador (pesos eslabones, momentos, entre otros) puede conllevar a que en el plano real las velocidades y aceleraciones resultantes estén por fuera de los límites del manipulador.

La forma como se ha validado el método, la cual consiste en trayectorias de salida almacenadas en hojas de cálculo, que poseen información de los parámetros cinemáticos, diferenciales y dinámicos de cada punto del recorrido que la conforman, permite realizar análisis a fondo y comparar entre trayectorias el comportamiento del criterio que se considere importante para realizar análisis puntuales.

Este trabajo puede ser usado como punto de partida para proyectos de investigación tanto de pregrado como posgrado, enfocados en el área de robótica debido a la diversidad de temas tratados en el proyecto.

Entre estos temas uno de los principales a ahondar, serían los métodos de identificación de parámetros de manipuladores comerciales implementando los descritos en este trabajado, o investigando nuevos métodos que redujeran a lo más mínimo la intervención mecánica y eléctrica del manipulador.

La validación física de los datos de velocidades, torques y consumos por trayectoria que arroja el algoritmo en manipuladores comerciales mediante instrumentos de medición es otro tema de interés a desarrollar como proyecto de investigación.

Realizar investigaciones en las cuales se implementen y comparen otros métodos de optimización con el enfoque propuesto sería de gran importancia porque se podría llegar a identificar un método que reduzca el tiempo de cómputo del algoritmo.

Con los modelos de cálculo desarrollados se pueden generar proyectos de investigación enfocados en el control del manipulador ya sea por el criterio de torques, velocidades, aceleraciones o consumo eléctrico.

Como temas secundarios que pueden derivarse del proyecto se identifican proyectos de pregrado para realización de programas de generación de trayectorias, diseño mecánico de manipuladores antropomórficos de seis grados de libertad y muñeca esférica, y aplicación del método en otros tipos de procesos (ensamble, pintura, entre otros).

REFERENCIAS BIBLIOGRÁFICAS

ABICOR BINZEL. (2013). *Download*. Recuperado el 10 de Septiembre de 2013, de Binzel-Abicor: http://www.binzel-abicor.com/uploads/Content/Germany/PDF-Files/PDF_Files_ROBO/English/ROBO_PROR143_GB_40_WEB_PW.pdf

Albert, F., Koh, S., Chen, C., Tiong, S., & Edwin, S. (2009). Optimizing Joint Angles of Robotic Manipulator using genetic Algorithm. *ICCEA 2009 : International Conference on Computer Engineering and Applications* (págs. 134-139). Singapore: IPCSIT.

Altuzarra, O., Sandru, B., Macho, E., & Amezua, E. (2010). Análisis de indicadores cinemáticos para el diseño de manipuladores paralelos. *XVIII Congreso Nacional de Ingeniería Mecánica*. Ciudad Real.

Angeles, J. (2003). *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms* (2nd ed.). New York: Springer Verlag.

Armstrong, B., Khatib, O., & Burdick, J. (1986). The Explicit Dynamic Model and Inertial Parameters of the PUMA 566 Arm. *International Conference on Robotics and Automation* (págs. 510 - 518). San Francisco: IEEE.

Ayten, K. K., Iravani, P., & Sahinkaya, M. N. (2011). Optimum Trajectory Planning For Industrial Robots Through Inverse Dynamics. *8th International Conference on Informatics in Control, Automation and Robotics-ICINCO 2011*, (págs. 105-110). Setubal.

Briot, S., & Bonev, I. A. (2007). Are Parallel Robots More Accurate Than Serial Robots? *Author manuscript, published in "CSME Transactions" , 31 (4), 445-456*.

Compucraft Ltd. (2012). *RobotWorks - Robotic interface for SolidWorks*. Recuperado el 10 de Septiembre de 2013, de RobotWorks: <http://www.robotworks-eu.com/products/faq-Kuka.htm>

Corke, P. I. (1996). *Visual control of robots : high-performance visual servoing*. Research Studies Pre.

Corke, P. (1996). In situ Measurement of Robot Motor Electrical Constants. *Robotica , 14 (04), 433-436*.

Corke, P. (1994). *The Unimation Puma servo system*. Locked bag 9, Preston: CSIRO Division of Manufacturing Technology.

Craig, J. (2005). *Introduction to Robotics Mechanics and Control* (3rd ed.). New Jersey: Pearson Education International.

Dos Santos, R. R., Steffen Jr, V., & Pereira Saramago, S. d. (2006). Robot path planning in a constrained workspace by using optimal control techniques. *Multibody System Dynamics* , 19 (1-2), 159-177.

Dos Santos, R., Steffen Jr, V., & Pereira Saramago, S. d. (2010). Optimal Task Placement of a Serial Robot Manipulator for Manipulability and Mechanical Power Optimization. *Intelligent Information Management* , 2 (9), 512-525.

Fateh, M. M., Farahani, S. S., & Khatamianfar, A. (2010). Task Space Control of a Welding Robot Using a Fuzzy Coordinator. *International Journal of Control, Automation, and Systems* , 8 (3), 574-582.

González, C., Blanco, D., & Moreno, L. (2009). Optimum robot manipulator path generation using Differential Evolution. *Congress on Evolutionary Computation, 2009. CEC '09.* (págs. 3322 - 3329). Trondheim: IEEE.

Group, K. R. (2009). www.kuka-robotics.com/res/sps/f776ebab-f613-4818-9feb-527612db8dc4_PF0012_KR_5_arc_HW_en.pdf. Recuperado el 10 de Septiembre de 2013, de Kuka-Robotics: http://www.kuka-robotics.com/res/sps/f776ebab-f613-4818-9feb-527612db8dc4_PF0012_KR_5_arc_HW_en.pdf

Kamrani, B., Berbyuk, V., & Wäppling, D. (2009). Optimal robot placement using response surface method. *The International Journal Of Advanced Manufacturing Technology* , 44 (1-2), 201-210.

Khoukhi, A., Baron, L., & Balazinski, M. (2009). Constrained multi-objective trajectory planning of parallel kinematic machines. *Robotics and Computer-Integrated Manufacturing* , 25, 756–769.

Lanzutti, A. (2011). Smooth Trajectory Planning Algorithms For Industrial Robots: An Experimental Evaluation. *Annals of Faculty Engineering Hunedoara – International Journal of Engineering* , 127-132.

Liao, X., Wang, W., Lin, Y., & Gong, C. (2010). Time-optimal trajectory planning for a 6R jointed welding robot using adaptive genetic algorithms. *International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE)*. 2, págs. 600-603. Changchun: IEEE.

Meike, D., & Ribickis, L. (2011). Industrial Robot Path Optimization Approach with Asynchronous FLY-BY in Joint Space. *International Symposium on Industrial Electronics (ISIE, 2011)* (págs. 911 - 915). Gdansk: IEEE.

NA, H., Kim, I., Kang, B., & Shim, J. (2011). A experiment study for welding optimization of fillet welded structure. *Journal of Achievements in Materials and Manufacturing Engineering* , 178-187.

Panneerselvam, R. (2012). *Production and Operations Management* (3rd ed.). New Delhi: PHI Learning Private Limited.

Pires, J. (2007). *Industrial Robots Programming. Building Applications for the Factories of the Future* (2nd ed.). Springer Verlag.

Pires, J., Loureiro, A., & Bölmsjö, G. (2006). *Welding Robots, Technology, System Issues and Applications*. Springer-Verlag.

Ramírez, J., Jaramillo, J., & Duque, L. (2012). Manipulability index study on the KUKA robot KR5 ARC HW. *Signal Processing, and Artificial Vision (STSIVA), 2012 XVII Symposium of Image* (págs. 72-77). Medellín: IEEE.

Saunders, H. (1997). *MIG/MAG Welding Guide For Gas Metal Arc Welding (GMAW)*. Cleveland: American Society for Metals.

Sharma, G. S., & Kaur, A. (2011). Optimization of Energy in Robotic arm using Genetic Algorithm. *International Journal of Computer Science and Technology* , 2 (2), 315-317.

Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2009). *Robotics: Modelling, Planning and Control*. London: Springer Verlag.

Son, J., Kim, I., Kim, H., Kim, I., Kang, B., & Kim, H. (2007). A study on the prediction of bead geometry in the robotic welding system. *Mechanical Science and Technology* , 1726-1731.

Spong, M., Hutchinson, S., & Vidyasagar, M. (2004). *Robot Dynamics and Control* (2nd ed.). New York: John Wiley & Sons.

Spong, M., Hutchinson, S., & Vidyasagar, M. (2005). *Robot Modeling and Control*. New York: John Wiley and Sons, Inc.

Toscano, R. (2010). Stochastic Methods for Hard Optimization Application to Robust Fault Diagnosis and Control of Industrial Systems. En G. Rigatos, *Intelligent Industrial Systems: Modeling, Automation and Adaptive Behavior* (págs. 182-220). Information Science Reference.

Toscano, R., & Lyonnet, P. (2010). A new heuristic approach for non-convex optimization problems. *Information Sciences* , 1955–1966.

Toscano, R., & Lyonnet, P. (2009). Heuristic Kalman Algorithm for Solving Optimization Problems. *IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics* , 39 (5), 1231 - 1244.

Tregaskiss. (2012). *TOUGH GUN G2 Series Robotic MIG Guns*. Recuperado el 10 de Septiembre de 2013, de Tregaskiss: http://www.tregaskiss.com/files/M074_TOUGH_GUN_G2_Series_Robotic_MIG_Guns.pdf

UNIMATION. (1984). *UNIMATE PUMA Series 500 Industrial Robot*. Westinghouse Company.

Ur-Rehman, R., Caro, S., Chablat, D., & Wenger, P. (2010). Multiobjective Path Placement Optimization of Parallel Kinematics Machines Based on Energy Consumption, Shaking Forces and

Maximum Actuators Torques: Application to the Orthoglide. *Mechanism and Machine Theory* , 8 (45), 1125-1141.

Yoshikawa, T. (1987). Analysis and Control of Robot Manipulators with Redundancy. *Robotics Research The First International Symposium* , 735-747.

Yoshikawa, T. (1985). Manipulability of Robot Mechanisms. *The International Journal of Robotics Research* , 4, 3-9.

Yoshikawa, T. (1991). Translational and rotational manipulability of robotic manipulators. *Proceedings of the 1991 International Conference on Industrial Electronics, Control, and Instrumentation* (págs. 1170-1175). Kobe: IEEE.

Zhang, D., & Qi, L. (2008). Virtual Engineering: Optimal Cell Layout Method for Improving Productivity for Industrial Robot. *2008 IEEE Conference on Robotics, Automation and Mechatronics (RAM)* (págs. 6 - 11). Chengdu: IEEE.

ANEXOS

ANEXO 1: PSEUDOCODIGO ALGORITMO PLANEACIÓN DE TRAYECTORIAS

En esta sección se expone el pseudocódigo del algoritmo de planeación de trayectorias para manipuladores seriales antropomórficos de seis grados y libertad y muñeca esférica basado en los criterios de minimización del consumo eléctrico y la maximización de la manipulabilidad. Cabe aclarar que solo se expondrán solo las funciones propias del algoritmo y no se ahondará en declaración de variables, funciones básicas, como por ejemplo las relacionadas con operaciones entre matrices, y en rutinas para evitar la singularidad del algoritmo, aunque si fueron implementadas en el algoritmo final.

```
//ALGORITMO DE OPTIMIZACIÓN DE TRAYECTORIAS PARA MANIPULADORES//  
//SERIALES ANTROPOMÓRFICOS DE SEIS GRADOS DE LIBERTAD Y MUÑECA ESFÉRICA//  
//BASADOS EN LOS CRITERIOS DE MINIMIZACIÓN DEL CONSUMO ELÉCTRICO Y//  
//MAXIMIZACIÓN DE LA MANIPULABILIDAD//
```

FUNCIONES ALGORITMO OPTIMIZACIÓN DE TRAYECTORIAS

```
//DECLARACION FUNCIONES//
```

Función **DHMatriz**(Theta, di, ai, alpha) Devolver (DHSalida())

```
//
```

```
//FUNCION PARA HALLAR PARAMETROS DH//
```

```
//
```

```
    Tzt(1, 1) = Cos(Theta)
```

```
    Tzt(1, 2) = -Sen(Theta)
```

```
    Tzt(1, 3) = 0
```

```
    Tzt(1, 4) = 0
```

```
    Tzt(2, 1) = Sen(Theta)
```

```
    Tzt(2, 2) = Cos(Theta)
```

```
    Tzt(2, 3) = 0
```

```
    Tzt(2, 4) = 0
```

```
    Tzt(3, 1) = 0
```

```
    Tzt(3, 2) = 0
```

```
    Tzt(3, 3) = 1
```

```
    Tzt(3, 4) = 0
```

```
    Tzt(4, 1) = 0
```

```
    Tzt(4, 2) = 0
```

```
    Tzt(4, 3) = 0
```

```
    Tzt(4, 4) = 1
```

```
    Tdi(1, 1) = 1
```

Tdi(1, 2) = 0
Tdi(1, 3) = 0
Tdi(1, 4) = 0
Tdi(2, 1) = 0
Tdi(2, 2) = 1
Tdi(2, 3) = 0
Tdi(2, 4) = 0
Tdi(3, 1) = 0
Tdi(3, 2) = 0
Tdi(3, 3) = 1
Tdi(3, 4) = di
Tdi(4, 1) = 0
Tdi(4, 2) = 0
Tdi(4, 3) = 0
Tdi(4, 4) = 1

Tai(1, 1) = 1
Tai(1, 2) = 0
Tai(1, 3) = 0
Tai(1, 4) = ai
Tai(2, 1) = 0
Tai(2, 2) = 1
Tai(2, 3) = 0
Tai(2, 4) = 0
Tai(3, 1) = 0
Tai(3, 2) = 0
Tai(3, 3) = 1
Tai(3, 4) = 0
Tai(4, 1) = 0
Tai(4, 2) = 0
Tai(4, 3) = 0
Tai(4, 4) = 1

Txaph(1, 1) = 1
Txaph(1, 2) = 0
Txaph(1, 3) = 0
Txaph(1, 4) = 0
Txaph(2, 1) = 0
Txaph(2, 2) = Cos(alpha)
Txaph(2, 3) = -Sen(alpha)
Txaph(2, 4) = 0
Txaph(3, 1) = 0
Txaph(3, 2) = Sen(alpha)
Txaph(3, 3) = Cos(alpha)
Txaph(3, 4) = 0
Txaph(4, 1) = 0
Txaph(4, 2) = 0
Txaph(4, 3) = 0

```

Txaph(4, 4) = 1

DHSalida = Tzt() * Tdi() * Tai() * Txaph()
Fin Función DHMatriz
Función Cinem_Direct(d_1, a_1, alfa_1, d_2, a_2, alfa_2, d_3, a_3, alfa_3, d_4, a_4, alfa_4, d_5,
a_5, alfa_5, d_6, a_6, alfa_6, L_ART()) Devolver (Mat_CDirecta())
//
//FUNCION CALCULO CINEMATICA DIRECTA//
//
//Cálculo Matrices DH De Cada Eslabón//
A01 = DHMatriz(Theta1, d_1, a_1, alfa_1)
A12 = DHMatriz(Theta2, d_2, a_2, alfa_2)
A23 = DHMatriz(Theta3, d_3, a_3, alfa_3)
A34 = DHMatriz(Theta4, d_4, a_4, alfa_4)
A45 = DHMatriz(Theta5, d_5, a_5, alfa_5)
A56 = DHMatriz(Theta6, d_6, a_6, alfa_6)

//Cálculo Matrices De Transformación Homogénea//
T02 = MultMatriz(A01(), A12())
T03 = MultMatriz(T02(), A23())
T04 = MultMatriz(T03(), A34())
T05 = MultMatriz(T04(), A45())
T06 = MultMatriz(T05(), A56())

Mat_CDirecta = [A01() A12() A23() A34() A45() A56() T02() T03() T04()
T05() T06()]T
//
Fin Función Cinem_Direct

Funcion Espacio_Trab(d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h,
L_ART()) Devolver (Mat_Salida())
//
//FUNCION PARA CALCULAR ESPACIO ESFERICO APROXIMADO DE ROBOT//
//
//Límites Superiores Espacio Esférico Aproximado//
Theta1 ← 0
Theta2 ← L_ART(2, 2)
Theta3 ← (L_ART(3, 2) + L_ART(3, 1)) / 2
Theta4 ← Pi1
Theta5 ← 0
Theta6 ← 0

//Llama función para cálculo de Cinemática Directa //
Mat_CDirecta ← Cinem_Direct(Theta1, d_1, a_1, alfa_1, Theta2, 0, a_2, 0, Theta3, d_3,
a_3, alfa_3, Theta4, d_4, 0, alfa_4, Theta5, 0, 0, alfa_5, Theta6, d_6+d_h, a_h, 0)
P04x ← Mat_CDirecta(33, 4)
P04z ← Mat_CDirecta(35, 4)

```

```

P01x ← Mat_CDirecta(1, 4)
P01z ← Mat_CDirecta(3, 4)
Radio_May ← Sqr((P04x - P01x + a_1) ^ 2 + (P04z - P01z) ^ 2)

//Calcula Ángulos de Colatitud inicial y Final Espacio de Trabajo//
Ang_Cini ← -Pi1 / 2
Ang_Cfin ← ACos(P04x / Radio_May)

//Límites Inferiores Espacio Esférico Aproximado//
Theta2 ← -Pi1 / 2
Theta3 ← L_ART(3, 2)

//Llama función para cálculo de Cinemática Directa //
Mat_CDirecta ← Cinem_Direct(Theta1, d_1, a_1, alfa_1, Theta2, 0, a_2, 0, Theta3, d_3,
a_3, alfa_3, Theta4, d_4, 0, alfa_4, Theta5, 0, 0, alfa_5, Theta6, d_6+d_h, a_h, 0)
P04x ← Mat_CDirecta(33, 4)
P04z ← Mat_CDirecta(35, 4)
Radio_Men ← Sqr((P04x) ^ 2 + (P04z - d_1) ^ 2)

//Calcula Ángulos Azimutales inicial y Final Espacio de Trabajo//
Ang_Aini ← L_ART(1, 1)
Ang_Afin ← L_ART(1, 2)

Mat_Salida(1, 1) ← Radio_Men
Mat_Salida(2, 1) ← Radio_May
Mat_Salida(3, 1) ← Ang_Cini
Mat_Salida(4, 1) ← Ang_Cfin
Mat_Salida(5, 1) ← Ang_Aini
Mat_Salida(6, 1) ← Ang_Afin
//
Fin Función Espacio_Trab

Función Cinem_Inver (pX_, pY_, pZ_, APrec, ANut, ARotP, d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3,
d_4, alfa_4, alfa_5, d_6, d_h, a_h) Devolver (Vec_CInv())
//
//FUNCIÓN CÁLCULO DE CINEMÁTICA INVERSA DE UN PUNTO XYZ CON ORIENTACIÓN APREC,
ANUT, AROTP//
//
//Vector "n" Matriz Rotación Efecto Final//
nx_I ← Cos(ARotP) * Cos(APrec) - Cos(ANut) * Sen(ARotP) * Sen(APrec)
ny_I ← Cos(ARotP) * Sen(APrec) + Cos(ANut) * Cos(ARotP) * Sen(APrec)
nz_I ← Sen(ANut) * Sen(ARotP)

//Vector "a" Matriz Rotación Efecto Final//
ax_I ← Sen(ANut) * Sen(APrec)
ay_I ← -Sen(ANut) * Cos(APrec)
az_I ← Cos(ANut)

```

```

//Punto Ow Ubicación Muñeca Esférica//
Owx ← pX_ + (d_6 + d_h) * ax_l + nx_l * a_h
Owy ← pY_ + (d_6 + d_h) * ay_l + ny_l * a_h
Owz ← pZ_ + (d_6 + d_h) * az_l + nz_l * a_h

//Distancia Horizontal (dm_) De a_1 Hasta Punto Ow Muñeca//
dm_ ← RAIZ(Owx ^ 2 + Owy ^ 2 - d_3 ^ 2) - a_1

//Distancia Vertical (dn_) De d_1 Hasta punto Ow Muñeca//
dn_ ← Owz - d_1

//Distancia Total (dR_) De a_1 Hasta Punto Ow Muñeca//
dR_ ← RAIZ(dm_ ^ 2 + dn_ ^ 2)

//Distancia Diagonal (h_34) a_3 de d_4//
h_34 ← RAIZ(a_3 ^ 2 + d_4 ^ 2)

//Angulo fi_2 Entre dR_ y dm_//
fi_2 ← Atan2(dn_, dm_)

//Angulo af_2 Entre dR_ y a_2//
af_2 ← ACos ((dR_ ^ 2 + a_2 ^ 2 - h_34 ^ 2) / (2 * a_2 * dR_))

//Angulo fi_3 Entre a_2 y h_34//
fi_3 ← ACos ((a_2 ^ 2 + h_34 ^ 2 - dR_ ^ 2) / (2 * a_2 * h_34))

//Angulo af_3 Entre dR_ y a_2//
af_3 ← Atan2(d_4, a_3)

//Ángulos Theta 1 A Theta 3 Hallados Con Cinemática Inversa//
Theta1l ← Atan2(Owy, Owx) - ASeno(d_3 / RAIZ(Owx ^ 2 + Owy ^ 2))
Theta2l ← -(fi_2 + af_2)
Theta3l ← Pi1 - fi_3 - af_3

//Valores Matriz De Rotación R03 Hallados Con Cinemática Inversa//
R03_11 ← Cos(Theta1l) * Cos(Theta2l + Theta3l)
R03_21 ← Sen(Theta1l) * Cos(Theta2l + Theta3l)
R03_31 ← -(Sen(Theta2l + Theta3l))
R03_13 ← -Cos(Theta1l) * Sen(Theta2l + Theta3l)
R03_23 ← -Sen(Theta1l) * Sen(Theta2l + Theta3l)
R03_33 ← -Cos(Theta2l + Theta3l)

//Valores Inversa Matriz De Rotación R03 Hallados Con Cinemática Inversa//
r13 ← R03_11 * ax_l + R03_21 * ay_l + R03_31 * az_l
r31 ← R03_13 * nx_l + R03_23 * ny_l + R03_33 * nz_l
r33 ← R03_13 * ax_l + R03_23 * ay_l + R03_33 * az_l

```

```

//Ángulos Theta 4 A Theta 6 Hallados Con Cinemática Inversa//
Theta4I ← ACos(-(r13 / (RAIZ(1 - r33 ^ 2))))
Theta5I ← (ACos(r33) - Pi1)
Theta6I ← ACos(r31 / (RAIZ(1 - r33 ^ 2)))

Vec_CInv(1, 1) ← pX_
Vec_CInv(1, 2) ← pY_
Vec_CInv(1, 3) ← pZ_
Vec_CInv(1, 4) ← APrec
Vec_CInv(1, 5) ← ANut
Vec_CInv(1, 6) ← ARotP
Vec_CInv(1, 7) ← Theta1I
Vec_CInv(1, 8) ← Theta2I
Vec_CInv(1, 9) ← Theta3I
Vec_CInv(1, 10) ← Theta4I
Vec_CInv(1, 11) ← Theta5I
Vec_CInv(1, 12) ← Theta6I

//
Fin Función Cinem_Inver

Función Criterios_optL(Vec_CInv(),d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6,
d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ, AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(),
In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(), Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)
Devolver (Vec_Salida())

//
//FUNCIÓN CÁLCULO CRITERIOS DE OPTIMIZACIÓN MANIPULABILIDAD Y CONSUMO ELÉCTRICO DE
UN PUNTO XYZ CON ORIENTACIÓN APREC, ANUT,AROTP //
//
Theta1 ← Vec_CInv(1, 7)
Theta2 ← Vec_CInv(1, 8)
Theta3 ← Vec_CInv(1, 9)
Theta4 ← Vec_CInv(1, 10)
Theta5 ← Vec_CInv(1, 11)
Theta6 ← Vec_CInv(1, 12)

//Almacena Valores Entrada En Vector Salida//
Vec_Salida(1, 2) ← Vec_CInv(1, 1)
Vec_Salida(1, 3) ← Vec_CInv(1, 2)
Vec_Salida(1, 4) ← Vec_CInv(1, 3)
Vec_Salida(1, 5) ← Vec_CInv(1, 4)
Vec_Salida(1, 6) ← Vec_CInv(1, 5)
Vec_Salida(1, 7) ← Vec_CInv(1, 6)

//Rutina Para Almacenar Valores De Articulaciones Theta y Comprobación De Límites De
Estas En Vector De Salida//
Para i ← 1 Hasta 6 Hacer
    Vec_Salida(1, 7 + i) ← Vec_CInv(1, 6 + i) //Thetai//

```

```

        Si (Vec_CInv(1, 6 + i) <= L_ART(i, 2)) Y (Vec_CInv(1, 6 + i) >< L_ART(i, 1)) Entonces
            Vec_Salida(1, i + 13) ← 0
        Sino
            Vec_Salida(1, i + 13) ← 1
        Fin_Si
    Fin_Para
    //
//SECCION CALCULO CINEMATICA DIRECTA//
    //
    //Cálculo Matrices DH De Cada Eslabón//
    A01 ← DHMatriz(Theta1, d_1, a_1, alfa_1)
    A12 ← DHMatriz(Theta2, 0, a_2, 0)
    A23 ← DHMatriz(Theta3, d_3, a_3, alfa_3)
    A34 ← DHMatriz(Theta4, d_4, 0, alfa_4)
    A45 ← DHMatriz(Theta5, 0, 0, alfa_5)
    A56 ← DHMatriz(Theta6, d_6+d_h, a_h, 0)

    //Cálculo Matrices De Transformación Homogénea//
    T02 ← A01() * A12()
    T03 ← T02() * A23()
    T04 ← T03() * A34()
    T05 ← T04() * A45()
    T06 ← T05() * A56()
    //
//SECCION CALCULO CINEMATICA DIFERENCIAL//
    //
    //Cálculo Matrices De Rotación//
    Q1 ← A01(1:3, 1:3)
    Q2 ← A12(1:3, 1:3)
    Q3 ← A23(1:3, 1:3)
    Q4 ← A34(1:3, 1:3)
    Q5 ← A45(1:3, 1:3)
    Q6 ← A56(1:3, 1:3)
    Q1Q2 ← T02(1:3, 1:3)
    Q1Q2Q3 ← T03(1:3, 1:3)
    Q1Q2Q3Q4 ← T04(1:3, 1:3)
    Q1Q2Q3Q4Q5 ← T05(1:3, 1:3)

    //Vectores De Posición//
    //Vector en marco de referencia del eslabón i//
    a1_1 ← A01(1:3, 4)
    a2_2 ← A12(1:3, 4)
    a3_3 ← A23(1:3, 4)
    a4_4 ← A34(1:3, 4)
    a5_5 ← A45(1:3, 4)
    a6_6 ← A56(1:3, 4)
    //Vector en marco de referencia base//
    a2_1 ← Q1() * a2_2()

```

```

a3_1 ← Q1Q2() * a3_3()
a4_1 ← Q1Q2Q3() * a4_4()
a5_1 ← Q1Q2Q3Q4() * a5_5()
a6_1 ← Q1Q2Q3Q4Q5() * a6_6()

//zi Vector Unitario Dirección Asociado Con El Eje De La Articulación//
z1_1(1, 1) ← 0
z1_1(2, 1) ← 0
z1_1(3, 1) ← 1
z2_1 ← Q1() * z1_1()
z3_1 ← Q1Q2() * z1_1()
z4_1 ← Q1Q2Q3() * z1_1()
z5_1 ← Q1Q2Q3Q4() * z1_1()
z6_1 ← Q1Q2Q3Q4Q5() * z1_1()

//Vectores Oi Hasta Ow//
pw3_1 ← a3_1() + a4_1()
pw2_1 ← a2_1() + pw3_1()
pw1_1 ← a1_1() + pw2_1()

//Producto Cruz Para Manipulabilidad, (Hasta Punto Ow)//
z1xpw1_1 ← z1_1() x pw1_1()
z2xpw2_1 ← z2_1() x pw2_1()
z3xpw3_1 ← z3_1()x pw3_1()

//Jacobiana Brazo Translacional (JTA)//
JTA ← [z1xpw1_1()   z2xpw2_1()   z3xpw3_1()]

//Manipulabilidad Sección Brazo wA//
wA_ ← Det(JTA())

//Jacobiana Muñeca (JRW)//
JRW ← [z4_1() z5_1() z6_1()]

//Transpuesta JRW//
JRW_T ← (JRW())T

//Manipulabilidad wW//
JRW_JRW_T ← JRW() * JRW_T()
wW_ ← (RAIZ(Det (JRW_JRW_T())))

//Manipulabilidad Total//
w_ ← wA_ * wW_

//Jacobiana Brazo Rotacional (JRA)//
JRA ← [z1_1() z2_1() z3_1()]

//Valores Velocidades Efecto Final//

```

```

MatVeIEF(1, 1) ← WX
MatVeIEF(2, 1) ← WY
MatVeIEF(3, 1) ← WZ
MatVeIEF(4, 1) ← VelX
MatVeIEF(5, 1) ← VelY
MatVeIEF(6, 1) ← VelZ

//Matrices De Producto Cruz De Vector Efecto Final Y Muñeca//
//Vector Efecto Final respecto a Origen Robot//
VEf ← T06(1: 3, 4)
//Matriz Antisimétrica de Vector Efecto Final//
MatVEf ← (VEf(x))

//Vector Muñeca respecto a Origen Robot//
VMc ← T05( 1:3, 4)

//Matriz Antisimétrica de Vector Muñeca
MatVMc ← (VMc(x))

//Resta Matrices Efecto Final y Muñeca//
MVEf_MVMc ← MatVEf() - MatVMc()

//Matriz de Estructura Triangular Inferior(U)//
Para i ← 1 Hasta 3 Hacer
    Para j ← 1 Hasta 3 Hacer
        zeros(i, j) ← 0
    Fin_Para
Fin_Para

Para i ← 1 To 3
    Para j ← 1 Hasta 3 Hacer
        Si i = j Entonces
            I1_1(i, j) ← 1
        Sino
            I1_1(i, j) ← 0
        Fin_Si
    Fin_Para
Fin_Para

MatU6_6 ← [I1_1() zeros()]
MatU6_6b ← [MVEf_MVMc() I1_1()]
MatU6_6 ← [MatU6_6() MatU6_6b()]T

//Matrices Velocidades Muñeca//
MatVelWr ← MatU6_6() * MatVeIEF()
MatVelWr_w ← MatVelWr( 1:3, 1) //Velocidades angulares muñeca//
MatVelWr_L ← MatVelWr( 4:6, 1) //Velocidades lineales muñeca//

```

```

//Calculo Inversas De Jacobianas Para Manipulador Con Muñeca Esférica//
//Inversa JTA//
INV_JTA ← (JTA())-1

//Inversa JRW//
INV_JRW ← (JRW())-1

//Matrices Velocidades Articulares Sección Brazo (dTheta 1 a 3)//
dqA ← INV_JTA() * MatVelWr_L()

//Matrices Velocidades Articulares Sección Muñeca (dTheta 4 a 6)//
dqW ← INV_JRW() * [MatVelWr_w() - (JRA() * dqA())]

//Valores Aceleraciones Efecto Final//
MatAcelEF(1, 1) ← AcWX
MatAcelEF(2, 1) ← AcWY
MatAcelEF(3, 1) ← AcWZ
MatAcelEF(4, 1) ← AcX
MatAcelEF(5, 1) ← AcY
MatAcelEF(6, 1) ← AcZ

//Matrices Aceleraciones Muñeca//
MatAcelWr ← MatU6_6() * MatAcelEF()
MatAcelWr_w ← (MatAcelWr( 1: 3, 1) //Aceleraciones angulares muñeca//
MatAcelWr_L ← MatAcelWr( 4:6, 1) // Aceleraciones lineales muñeca//

dT1Z1_1 ← z1_1() * dqA(1, 1)
dT2Z2_1 ← z2_1() * dqA(2, 1)
dT3Z3_1 ← z3_1() * dqA(3, 1)
dT4Z4_1 ← z4_1() * dqW(1, 1)
dT5Z5_1 ← z5_1() * dqW(2, 1)
dT6Z6_1 ← z6_1() * dqW(3, 1)

//Velocidades angulares eslabones respecto al marco referencia base//
w1_1 ← dT1Z1_1
w2_1 ← dT2Z2_1() + w1_1()
w3_1 ← dT3Z3_1() + w2_1()
w4_1 ← dT4Z4_1() + w3_1()
w5_1 ← dT5Z5_1() + w4_1()
w6_1 ← dT6Z6_1() + w5_1()

//Derivadas Vectores z//
dz1_1 ← w1_1() x z1_1()
dz2_1 ← w2_1() x z2_1()
dz3_1 ← w3_1() x z3_1()
dz4_1 ← w4_1() x z4_1()
dz5_1 ← w5_1() x z5_1()

```

```

dz6_1 ← w6_1() x z6_1()

//Derivadas JRA Y JRW//
dJRA ← [dz1_1()      dz2_1()      z3_1()]
dJRW ← [dz4_1()      dz5_1()      dz6_1()]

w4xa4_1 ← w4_1() x a4_1()
w3xa3_1 ← w3_1() x a3_1()
w2xa2_1 ← w2_1() x a2_1()
w1xa1_1 ← w1_1() x a1_1()
dpw3_1 ← w3xa3_1() + w4xa4_1()
dpw2_1 ← w2xa2_1() + dpw3_1()
dpw1_1 ← w1xa1_1() + dpw2_1()
dz1xpw1_1 ← dz1_1() x pw1_1()
dz2xpw2_1 ← dz2_1() x pw2_1()
dz3xpw3_1 ← dz3_1() x pw3_1()
z1xdpw1_1 ← z1_1() x dpw1_1()
z2xdpw2_1 ← z2_1() x dpw2_1()
z3xdpw3_1 ← z3_1() x dpw3_1()

du1_1 ← dz1xpw1_1() + z1xdpw1_1()
du2_1 ← dz2xpw2_1() + z2xdpw2_1()
du3_1 ← dz3xpw3_1() + z3xdpw3_1()

//Derivada JTA//
dJTA ← [du1_1()      du2_1()      du3_1()]

//Matrices Aceleraciones Articulares Sección Brazo (ddTheta 1 a 3) //
ddqA ← INV_JTA() * [MatAcelWr_L() - (dJTA()* dqA)]

//Matrices Aceleraciones Articulares Sección Muñeca (ddTheta 4 a 6) //
ddqW ← INV_JRW() * [MatAcelWr_w() - dJRA() * dqA() - dJRW() * dqW() - JRA() * ddqA()]

//
//SECCION DINAMICA INVERSA//
//
//Matrices De Rotación Del Sistema De Coordinadas i-ésimo al (i-1)-ésimo//
Q1T ← (Q1())T
Q2T ← (Q2())T
Q3T ← (Q3())T
Q4T ← (Q4())T
Q5T ← (Q5())T
Q6T ← (Q6())T

//Matrices Velocidades Articulares Sección Brazo (dTheta 1 a 3) //
//Recálculo Velocidades Angulares Eslabones respecto al marco referencia base//

Para i ← 1 Hasta 3 Hacer

```

```

w_0(i, 1) ← 0
Fin_Para

w_1 ← Q1T() * (w_0() + z1_1() * dqA(1, 1))
w_2 ← Q2T() * (w_1() + z1_1() * dqA(2, 1))
w_3 ← Q3T() * (w_2() + z1_1() * dqA(3, 1))
w_4 ← Q4T() * (w_3() + z1_1() * dqW(1, 1))
w_5 ← Q5T() * (w_4() + z1_1() * dqW(2, 1))
w_6 ← Q6T() * (w_5() + z1_1() * dqW(3, 1))

//Aceleraciones Angulares Eslabones respecto al marco referencia base //
AA_0 ← w_0()
AA_1 ← Q1T() * [AA_0() + z1_1() * ddqA(1, 1) + w_0() x (z1_1() * dqA(1, 1))]
AA_2 ← Q2T() * [AA_1() + z1_1() * ddqA(2, 1) + w_1() x (z1_1() * dqA(2, 1))]
AA_3 ← Q3T() * [AA_2() + z1_1() * ddqA(3, 1) + w_2() x (z1_1() * dqA(3, 1))]
AA_4 ← Q4T() * [AA_3() + z1_1() * ddqW(1, 1) + w_3() x (z1_1() * dqW(1, 1))]
AA_5 ← Q5T() * [AA_4() + z1_1() * ddqW(2, 1) + w_4() x (z1_1() * dqW(2, 1))]
AA_6 ← Q6T() * [AA_5() + z1_1() * ddqW(3, 1) + w_5() x (z1_1() * dqW(3, 1))]

//Vector De Localización de (xi,yi,zi) desde el origen de (xi-1,yi-1,zi-1)//
p0_1(1, 1) ← a_1
p0_1(2, 1) ← d_1 * Sen(alfa_1)
p0_1(3, 1) ← d_1 * Cos(alfa_1)
p1_2(1, 1) ← a_2
p1_2(2, 1) ← 0 * Sen(0)
p1_2(3, 1) ← 0 * Cos(0)
p2_3(1, 1) ← a_3
p2_3(2, 1) ← d_3 * Sen(alfa_3)
p2_3(3, 1) ← d_3 * Cos(alfa_3)
p3_4(1, 1) ← 0
p3_4(2, 1) ← d_4 * Sen(alfa_4)
p3_4(3, 1) ← d_4 * Cos(alfa_4)
p4_5(1, 1) ← 0
p4_5(2, 1) ← 0 * Sen(alfa_5)
p4_5(3, 1) ← 0 * Cos(alfa_5)
p5_6(1, 1) ← a_h
p5_6(2, 1) ← (d_6 + d_h) * Sen(0)
p5_6(3, 1) ← (d_6 + d_h) * Cos(0)

//Aceleración Lineal Eslabón (AL_i) //
gr_ ← 9.81 //m/seg^2//
AL_0 ← w_0()
AL_0(3, 1) ← gr_ //Se incluye efecto de gravedad en algoritmo//
AL_1 ← AA_1() x p0_1() + w_1() x (w_1() x p0_1()) + Q1T() * AL_0()
AL_2 ← AA_2() x p1_2() + w_2() x (w_2() x p1_2()) + Q2T() * AL_1()
AL_3 ← AA_3() x p2_3() + w_3() x (w_3() x p2_3()) + Q3T() * AL_2()
AL_4 ← AA_4() x p3_4() + w_4() x (w_4() x p3_4()) + Q4T() * AL_3()
AL_5 ← AA_5() x p4_5() + w_5() x (w_5() x p4_5()) + Q5T() * AL_4()

```

$$AL_6 \leftarrow AA_6() \times p5_6() + w_6() \times (w_6() \times p5_6()) + Q6T() * AL_5()$$

//Vector 3x1 De La Posición Del Centro De Masa De Los Eslabones//

//Centro masa eslabón 1//

$$s_1(1, 1) \leftarrow M_s(1, 1)$$

$$s_1(2, 1) \leftarrow M_s(1, 2) * \text{Cos}(\text{alfa}_1) - M_s(1, 3) * \text{Sen}(\text{alfa}_1)$$

$$s_1(3, 1) \leftarrow M_s(1, 2) * \text{Sen}(\text{alfa}_1) + M_s(1, 3) * \text{Cos}(\text{alfa}_1)$$

$$s_1 \leftarrow s_1() - p0_1()$$

//Centro masa eslabón 2//

$$s_2(1, 1) \leftarrow M_s(2, 1)$$

$$s_2(2, 1) \leftarrow M_s(2, 2) * \text{Cos}(0) - M_s(2, 3) * \text{Sen}(0)$$

$$s_2(3, 1) \leftarrow M_s(2, 2) * \text{Sen}(0) + M_s(2, 3) * \text{Cos}(0)$$

$$s_2 \leftarrow s_2() - p1_2()$$

//Centro masa eslabón 3//

$$s_3(1, 1) \leftarrow M_s(3, 1)$$

$$s_3(2, 1) \leftarrow M_s(3, 2) * \text{Cos}(\text{alfa}_3) - M_s(3, 3) * \text{Sen}(\text{alfa}_3)$$

$$s_3(3, 1) \leftarrow M_s(3, 2) * \text{Sen}(\text{alfa}_3) + M_s(3, 3) * \text{Cos}(\text{alfa}_3)$$

$$s_3 \leftarrow s_3() - p2_3()$$

//Centro masa eslabón 4//

$$s_4(1, 1) \leftarrow M_s(4, 1)$$

$$s_4(2, 1) \leftarrow M_s(4, 2) * \text{Cos}(\text{alfa}_4) - M_s(4, 3) * \text{Sen}(\text{alfa}_4)$$

$$s_4(3, 1) \leftarrow M_s(4, 2) * \text{Sen}(\text{alfa}_4) + M_s(4, 3) * \text{Cos}(\text{alfa}_4)$$

$$s_4 \leftarrow s_4() - p3_4()$$

//Centro masa eslabón 5//

$$s_5(1, 1) \leftarrow M_s(5, 1)$$

$$s_5(2, 1) \leftarrow M_s(5, 2) * \text{Cos}(\text{alfa}_5) - M_s(5, 3) * \text{Sen}(\text{alfa}_5)$$

$$s_5(3, 1) \leftarrow M_s(5, 2) * \text{Sen}(\text{alfa}_5) + M_s(5, 3) * \text{Cos}(\text{alfa}_5)$$

$$s_5 \leftarrow s_5() - p4_5()$$

//Centro masa eslabón 6//

$$s_6(1, 1) \leftarrow M_s(6, 1)$$

$$s_6(2, 1) \leftarrow M_s(6, 2) * \text{Cos}(0) - M_s(6, 3) * \text{Sen}(0)$$

$$s_6(3, 1) \leftarrow M_s(6, 2) * \text{Sen}(0) + M_s(6, 3) * \text{Cos}(0)$$

$$s_6 \leftarrow s_6() - p5_6()$$

//Centro masa herramienta//

$$s_{6p}(1, 1) \leftarrow M_s(7, 1)$$

$$s_{6p}(2, 1) \leftarrow M_s(7, 2)$$

$$s_{6p}(3, 1) \leftarrow -(d_h - M_s(7, 3))$$

//Aceleración centro de masa//

$$ACM_1 \leftarrow AA_1() \times s_1() + w_1() \times (w_1() \times s_1()) + AL_1()$$

$$ACM_2 \leftarrow AA_2() \times s_2() + w_2() \times (w_2() \times s_2()) + AL_2()$$

```

ACM_3 ← AA_3() x s_3() + w_3() x (w_3() x s_3()) + AL_3()
ACM_4 ← AA_4() x s_4() + w_4() x (w_4() x s_4()) + AL_4()
ACM_5 ← AA_5() x s_5() + w_5() x (w_5() x s_5()) + AL_5()
ACM_6 ← AA_6() x s_6() + w_6() x (w_6() x s_6()) + AL_6()
ACM_6p ← AA_6() x s_6p() + w_6() x (w_6() x s_6p()) + AL_6() //Dato herramienta//

```

//Fuerza En El Centro De Masa De Cada Elemento//

```

FCM_1 ← ACM_1() * M_m(1, 1)
FCM_2 ← ACM_2() * M_m(2, 1)
FCM_3 ← ACM_3() * M_m(3, 1)
FCM_4 ← ACM_4() * M_m(4, 1)
FCM_5 ← ACM_5() * M_m(5, 1)
FCM_6 ← ACM_6() * M_m(6, 1)
FCM_6p ← ACM_6p() * M_m(7, 1) //Dato herramienta//

```

//Par En El Centro De Masa De Cada Elemento//

```

PCM_1 ← ln_1() * AA_1() + w_1() x (ln_1() * w_1())
PCM_2 ← ln_2() * AA_2() + w_2() x (ln_2() * w_2())
PCM_3 ← ln_3() * AA_3() + w_3() x (ln_3() * w_3())
PCM_4 ← ln_4() * AA_4() + w_4() x (ln_4() * w_4())
PCM_5 ← ln_5() * AA_5() + w_5() x (ln_5() * w_5())
PCM_6 ← ln_6() * AA_6() + w_6() x (ln_6() * w_6())
PCM_6p ← IPist() * AA_6() + w_6() x (IPist() * w_6()) //Dato herramienta//

```

//Fuerzas Articulares//

```

FArt_6p ← FCM_6p() //Dato herramienta//
FArt_6 ← FCM_6()
FArt_5 ← Q5() * FArt_6() + FCM_5()
FArt_4 ← Q4() * FArt_5() + FCM_4()
FArt_3 ← Q3() * FArt_4() + FCM_3()
FArt_2 ← Q2() * FArt_3() + FCM_2()
FArt_1 ← Q1() * FArt_2() + FCM_1()

```

//p(i+1)_i es el vector 3x1 de localización del sistema de referencia//

```

p2_1 ← Q2T() * p0_1()
p3_2 ← Q3T() * p1_2()
p4_3 ← Q4T() * p2_3()
p5_4 ← Q5T() * p3_4()
p6_5 ← Q6T() * p4_5()

```

//Pares Articulares//

```

PArt_6 ← (p5_6() + s_6()) x FCM_6() + (p5_6() + s_6p()) x FCM_6p() + PCM_6() + PCM_6p()
PArt_5 ← Q6() * (PArt_6() + p6_5() x FArt_6()) + (p4_5() + s_5()) x FCM_5() + PCM_5()
PArt_4 ← Q5() * (PArt_5() + p5_4() x FArt_5()) + (p3_4() + s_4()) x FCM_4() + PCM_4()
PArt_3 ← Q4() * (PArt_4() + p4_3() x FArt_4()) + (p2_3() + s_3()) x FCM_3() + PCM_3()
PArt_2 ← Q3() * (PArt_3() + p3_2() x FArt_3()) + (p1_2() + s_2()) x FCM_2() + PCM_2()
PArt_1 ← Q2() * (PArt_2() + p2_1() x FArt_2()) + (p0_1() + s_1()) x FCM_1() + PCM_1()

```

```

//Fuerzas Y Pares En Articulaciones//
VecTemp ← (PArt_1())T * Q1T() * z1_1()
T_Art(1, 1) ← VecTemp(1, 1) + b_(1, 1) * dqA(1, 1)
VecTemp ← PArt_2()T * Q2T() * z1_1()
T_Art(2, 1) ← VecTemp(1, 1) + b_(2, 1) * dqA(2, 1)
VecTemp ← PArt_3()T * Q3T() * z1_1()
T_Art(3, 1) ← VecTemp(1, 1) + b_(3, 1) * dqA(3, 1)
VecTemp ← PArt_4()T * Q4T() * z1_1()
T_Art(4, 1) ← VecTemp(1, 1) + b_(4, 1) * dqW(1, 1)
VecTemp ← PArt_5()T * Q5T() * z1_1()
T_Art(5, 1) ← VecTemp(1, 1) + b_(5, 1) * dqW(2, 1)
VecTemp ← PArt_6()T * Q6T() * z1_1()
T_Art(6, 1) ← VecTemp(1, 1) + b_(6, 1) * dqW(3, 1)

```

```

//Torque Motores//
Tm(1, 1) ← T_Art(1, 1) / nr(1, 1)
Tm(2, 1) ← T_Art(2, 1) / nr(2, 1)
Tm(3, 1) ← T_Art(3, 1) / nr(3, 1)
Tm(4, 1) ← T_Art(4, 1) / nr(4, 1)
Tm(5, 1) ← T_Art(5, 1) / nr(5, 1)
Tm(6, 1) ← T_Art(6, 1) / nr(6, 1)

```

```

//Velocidad Angular Motores//
wm(1, 1) ← nr(1, 1) * dqA(1, 1)
wm(2, 1) ← nr(2, 1) * dqA(2, 1)
wm(3, 1) ← nr(3, 1) * dqA(3, 1)
wm(4, 1) ← nr(4, 1) * dqW(1, 1)
wm(5, 1) ← nr(5, 1) * dqW(2, 1)
wm(6, 1) ← nr(6, 1) * dqW(3, 1)

```

```

//Corriente Motores//
Im(1, 1) ← Tm(1, 1) / Kt(1, 1)
Im(2, 1) ← Tm(2, 1) / Kt(2, 1)
Im(3, 1) ← Tm(3, 1) / Kt(3, 1)
Im(4, 1) ← Tm(4, 1) / Kt(4, 1)
Im(5, 1) ← Tm(5, 1) / Kt(5, 1)
Im(6, 1) ← Tm(6, 1) / Kt(6, 1)

```

```

//Potencial Electromotriz//
Ve(1, 1) ← Ke(1, 1) * wm(1, 1)
Ve(2, 1) ← Ke(2, 1) * wm(2, 1)
Ve(3, 1) ← Ke(3, 1) * wm(3, 1)
Ve(4, 1) ← Ke(4, 1) * wm(4, 1)
Ve(5, 1) ← Ke(5, 1) * wm(5, 1)
Ve(6, 1) ← Ke(6, 1) * wm(6, 1)

```

```

//Perdida Potencia Resistiva//
PJ(1, 1) ← R_(1, 1) * Im(1, 1) ^ 2

```

```

PJ(2, 1) ← R_(2, 1) * Im(2, 1) ^ 2
PJ(3, 1) ← R_(3, 1) * Im(3, 1) ^ 2
PJ(4, 1) ← R_(4, 1) * Im(4, 1) ^ 2
PJ(5, 1) ← R_(5, 1) * Im(5, 1) ^ 2
PJ(6, 1) ← R_(6, 1) * Im(6, 1) ^ 2

//Potencia Para Producir Fuerza Electromotriz//
PEM(1, 1) ← Ve(1, 1) * Im(1, 1)
PEM(2, 1) ← Ve(2, 1) * Im(2, 1)
PEM(3, 1) ← Ve(3, 1) * Im(3, 1)
PEM(4, 1) ← Ve(4, 1) * Im(4, 1)
PEM(5, 1) ← Ve(5, 1) * Im(5, 1)
PEM(6, 1) ← Ve(6, 1) * Im(6, 1)

//Potencia Consumida Por Motores (Watt)//
Pm(1, 1) ← PJ(1, 1) + PEM(1, 1)
Pm(2, 1) ← PJ(2, 1) + PEM(2, 1)
Pm(3, 1) ← PJ(3, 1) + PEM(3, 1)
Pm(4, 1) ← PJ(4, 1) + PEM(4, 1)
Pm(5, 1) ← PJ(5, 1) + PEM(5, 1)
Pm(6, 1) ← PJ(6, 1) + PEM(6, 1)

//Potencia Total (Kw)//
PotTotal ← (Abs(Pm(1, 1)) + Abs(Pm(2, 1)) + Abs(Pm(3, 1)) + Abs(Pm(4, 1)) + Abs(Pm(5, 1))
+ Abs(Pm(6, 1))) / 1000
//
//SECCION ALMACENAMIENTO TRAYECTORIA CON CRITERIOS CALCULADOS EN MATRIZ DE
SALIDA//
//
Para i ← 1 Hasta 6 Hacer
//Rutina Para Almacenar Valores De Torque Articulaciones, Torque Motores Y Consumos
Motores En Vector De Salida//
    Vec_Salida(1, 37 + i) ← T_Art(i, 1)    //Torques articulaciones//
    Vec_Salida(1, 37 + 6 + i) ← Tm(i, 1)  //Torques motores//
    Vec_Salida(1, 37 + 12 + i) ← PT(i, 1) //Potencias motores//
Fin_Para

Vec_Salida(1, 56) ← PotTotal //Almacena consumo total motores en vector de salida//
//Verifica si potencia total menor que potencia nominal robot//
Si Abs(PotTotal) <= Pn Entonces
    Vec_Salida(1, 57) ← 0
Sino
    Vec_Salida(1, 57) ← 1
Fin_Si

Para i ← 1 Hasta 3 Hacer
//Rutina Para Almacenar Valores De Velocidades Articulares Y Comprobación De Estas En
Vector De Salida//

```

```

Vec_Salida(1, 19 + i) ← dqA(i, 1) //dqA (dTheta1 a dTheta3)//
Vec_Salida(1, 19 + 3 + i) ← dqW(i, 1) //dqw (dTheta4 a dTheta6)//
Si Abs(dqA(i, 1)) <= L_VART(i, 1) Entonces
  Vec_Salida(1, 19 + 6 + i) ← 0
Sino
  Vec_Salida(1, 19 + 6 + i) ← 1
Fin_Si
Si Abs(dqW(i, 1)) <= L_VART(i + 3, 1) Entonces
  Vec_Salida(1, 19 + 6 + 3 + i) ← 0
Sino
  Vec_Salida(1, 19 + 6 + 3 + i) ← 1
Fin_Si
//Rutina Para Almacenar Valores De Aceleraciones Articulares En Vector De Salida//
Vec_Salida(1, 19 + 12 + i) ← ddqA(i, 1) //ddqA (ddTheta1 a ddTheta3)//
Vec_Salida(1, 19 + 12 + 3 + i) ← ddqW(i, 1) //ddqA (ddTheta1 a ddTheta3)//
//Rutina para almacenar valores de coordenadas de cada articulación//
Vec_Salida(1, 70 + i) ← A01(i, 4)
Vec_Salida(1, 73 + i) ← T02(i, 4)
Vec_Salida(1, 76 + i) ← T03(i, 4)
Vec_Salida(1, 79 + i) ← T04(i, 4)
Vec_Salida(1, 82 + i) ← T05(i, 4)
Fin_Para
//
Fin Función Criterios_optL

```

```

Función Valid_Trayectoria(Trayectoria(), D_px, D_py, D_pzd_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3,
d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ, AcWX, AcWY,
AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(), Kt(), Ke(), R_(), nr(),
L_ART(), L_VART(), Pn) Devolver (Mat_Salida())
//
//Función Para Hallar y Validar Criterios De Optimización De Manipulabilidad Y Consumo Eléctrico
De Trayectoria//
//

```

```

//Dimensión de la matriz de trayectoria//
Fila ← LimiteSuperior(Trayectoria, 1)
Col ← LimiteSuperior (Trayectoria, 2)
//Encabezados Matriz de Salida//
Mat_Salida(1, 2) ← "Px"
Mat_Salida(1, 3) ← "Py"
Mat_Salida(1, 4) ← "Pz"
Mat_Salida(1, 5) ← "APrec"
Mat_Salida(1, 6) ← "ANut"
Mat_Salida(1, 7) ← "ARotP"
Mat_Salida(1, 8) ← "Theta1"
Mat_Salida(1, 9) ← "Theta2"
Mat_Salida(1, 10) ← "Theta3"
Mat_Salida(1, 11) ← "Theta4"

```

Mat_Salida(1, 12) ← "Theta5"
Mat_Salida(1, 13) ← "Theta6"
Mat_Salida(1, 14) ← "CT1"
Mat_Salida(1, 15) ← "CT2"
Mat_Salida(1, 16) ← "CT3"
Mat_Salida(1, 17) ← "CT4"
Mat_Salida(1, 18) ← "CT5"
Mat_Salida(1, 19) ← "CT6"
Mat_Salida(1, 20) ← "dTheta1"
Mat_Salida(1, 21) ← "dTheta2"
Mat_Salida(1, 22) ← "dTheta3"
Mat_Salida(1, 23) ← "dTheta4"
Mat_Salida(1, 24) ← "dTheta5"
Mat_Salida(1, 25) ← "dTheta6"
Mat_Salida(1, 26) ← "Cd1"
Mat_Salida(1, 27) ← "Cd2"
Mat_Salida(1, 28) ← "Cd3"
Mat_Salida(1, 29) ← "Cd4"
Mat_Salida(1, 30) ← "Cd5"
Mat_Salida(1, 31) ← "Cd6"
Mat_Salida(1, 32) ← "ddTheta1"
Mat_Salida(1, 33) ← "ddTheta2"
Mat_Salida(1, 34) ← "ddTheta3"
Mat_Salida(1, 35) ← "ddTheta4"
Mat_Salida(1, 36) ← "ddTheta5"
Mat_Salida(1, 37) ← "ddTheta6"
Mat_Salida(1, 38) ← "T1"
Mat_Salida(1, 39) ← "T2"
Mat_Salida(1, 40) ← "T3"
Mat_Salida(1, 41) ← "T4"
Mat_Salida(1, 42) ← "T5"
Mat_Salida(1, 43) ← "T6"
Mat_Salida(1, 44) ← "Tm1"
Mat_Salida(1, 45) ← "Tm2"
Mat_Salida(1, 46) ← "Tm3"
Mat_Salida(1, 47) ← "Tm4"
Mat_Salida(1, 48) ← "Tm5"
Mat_Salida(1, 49) ← "Tm6"
Mat_Salida(1, 50) ← "P1"
Mat_Salida(1, 51) ← "P2"
Mat_Salida(1, 52) ← "P3"
Mat_Salida(1, 53) ← "P4"
Mat_Salida(1, 54) ← "P5"
Mat_Salida(1, 55) ← "P6"
Mat_Salida(1, 56) ← "Pot(Kw)"
Mat_Salida(1, 57) ← "CP"
Mat_Salida(1, 58) ← "wA"
Mat_Salida(1, 59) ← "wW"

```

Mat_Salida(1, 60) ← "1-w"
Mat_Salida(1, 61) ← "|u|"
Mat_Salida(1, 62) ← "VelT"
Mat_Salida(1, 63) ← "Tiempo"
Mat_Salida(1, 64) ← "Consumo"

Mag_U ← Raíz((Trayectoria(2, 1) - Trayectoria(1, 1)) ^ 2 + (Trayectoria(2, 2) - Trayectoria(1, 2)) ^ 2 + (Trayectoria(2, 3) - Trayectoria(1, 3)) ^ 2)
Si Mag_U = 0 Entonces
    VelX ← 0
    VelY ← 0
    VelZ ← 0
Sino
    VelX ← VelS * (Trayectoria(2, 1) - Trayectoria(1, 1)) / Mag_U
    VelY ← VelS * (Trayectoria(2, 2) - Trayectoria(1, 2)) / Mag_U
    VelZ ← VelS * (Trayectoria(2, 3) - Trayectoria(1, 3)) / Mag_U
Fin_Si

//Llama Función Cinemática Inversa Para Cálculo De Valor Articuciones Para Punto Inicial De Trayectoria Y Almacena Resultado En Vect_CInversa//
Vect_CInversa ← Cinem_Inver(Trayectoria(1, 1) + D_px, Trayectoria(1, 2) + D_py, Trayectoria(1, 3) + D_pz, Trayectoria(1, 4), Trayectoria(1, 5), Trayectoria(1, 6), d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h)

//Llama Función Cálculo Criterios De Optimización Manipulabilidad Y Consumo Eléctrico Para punto inicial de trayectoria y almacena resultado en vector Punto_Cr()//
Punto_Cr ← Criterios_optL(Vect_CInversa(),d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ, AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(), Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)

//Sumatorias De Comprobación De Variables Para Definir Restricciones En Algoritmo De Optimización//
//Sumatoria Comprobación Articuciones//
sumaCTh1 ← Punto_Cr(1, 14)
sumaCTh2 ← Punto_Cr(1, 15)
sumaCTh3 ← Punto_Cr(1, 16)
sumaCTh4 ← Punto_Cr(1, 17)
sumaCTh5 ← Punto_Cr(1, 18)
sumaCTh6 ← Punto_Cr(1, 19)
//Sumatoria Comprobación Velocidades//
sumaCdTh1 ← Punto_Cr(1, 26)
sumaCdTh2 ← Punto_Cr(1, 27)
sumaCdTh3 ← Punto_Cr(1, 28)
sumaCdTh4 ← Punto_Cr(1, 29)
sumaCdTh5 ← Punto_Cr(1, 30)
sumaCdTh6 ← Punto_Cr(1, 31)
//Sumatoria Comprobación Potencia//

```

```

sumaCPot ← Punto_Cr(1, 57)
//Sumatoria Manipulabilidad//
suma1_w ← Punto_Cr(1, 60)
//Sumatoria Consumo en KWH//
SumaCons ← 0

//Almacena El Número Del Punto De Trayectoria En Matriz De Salida//
Mat_Salida(2, 1) ← 1

//Almacena El Valores Hallados Para El Punto 1 De La Trayectoria En Matriz De Salida//
Para i ← 2 Hasta 64 Hacer
    Mat_Salida(2, i) ← Punto_Cr(1, i)
Fin_Para

//
Para i ← 2 Hasta Fila Hacer
//CICLO PARA CALCULAR LOS CRITERIOS DE OPTIMIZACIÓN EN TODOS LOS PUNTOS DE LA
TRAYECTORIA//

//Llama Función Cinemática Inversa Para Cálculo De Valor Articulaciones Theta A Partir Del
Punto Dado Y Almacena Resultado En Vect_CInversa//
Vect_CInversa ← Cinem_Inver(Trayectoria(i, 1) + D_px, Trayectoria(i, 2) + D_py,
Trayectoria(i, 3) + D_pz, Trayectoria(i, 4), Trayectoria(i, 5), Trayectoria(i, 6), d_1, a_1,
alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h)

//Rutina Para Hallar Tiempo Mínimo De Duración De Movimiento Sin Caer En Velocidades
De Articulaciones Que Superen los límites de Velocidad//
//Magnitud desplazamiento entre puntos trayectoria//
Si i < Fila Entonces
    Mag_U ← Raíz((Trayectoria(i + 1, 1) - Trayectoria(i, 1)) ^ 2 + (Trayectoria(i + 1, 2) -
    Trayectoria(i, 2)) ^ 2 + (Trayectoria(i + 1, 3) - Trayectoria(i, 3)) ^ 2)
Fin_Si

//Tiempo de movimiento lineal de punto anterior al actual, con VelS inicial//
TiempoMovL ← Mag_U / VelS

//Matriz mínimo tiempo de movimiento en articulación según máxima velocidad de
articulación y desplazamiento de articulación//
//Mínimo tiempo Theta1//
MatTpoTh(1, 1) ← Abs(Vect_CInversa(1, 7) - Mat_Salida(i, 8)) / L_VART(1, 1)
//Mínimo tiempo Theta2//
MatTpoTh(1, 2) ← Abs(Vect_CInversa(1, 8) - Mat_Salida(i, 9)) / L_VART(2, 1)
//Mínimo tiempo Theta3//
MatTpoTh(1, 3) ← Abs(Vect_CInversa(1, 9) - Mat_Salida(i, 10)) / L_VART(3, 1)
//Mínimo tiempo Theta4//
MatTpoTh(1, 4) ← Abs(Vect_CInversa(1, 10) - Mat_Salida(i, 11)) / L_VART(4, 1)
//Mínimo tiempo Theta5//
MatTpoTh(1, 5) ← Abs(Vect_CInversa(1, 11) - Mat_Salida(i, 12)) / L_VART(5, 1)

```

```

//Mínimo tiempo Theta6//
    MatTpoTh(1, 6) ← Abs(Vect_CInversa(1, 12) - Mat_Salida(i, 13)) / L_VART(6, 1)

//Rutina Para Hallar Tiempo Mayor De Movimiento Entre Las Seis Articulaciones. Este
Debe Ser Menor Que Tiempo De Movimiento Lineal De Punto De Trayectoria//
    TpoCritico ← MatTpoTh(1, 1)
    Para tp ← 2 Hasta 6 Hacer
        Si TpoCritico < MatTpoTh(1, tp) Entonces
            TpoCritico ← MatTpoTh(1, tp)
        Fin_Si
    Fin_Para

    Si TiempoMovL < TpoCritico Entonces
        TiempoTray ← TpoCritico //Tiempo Entre Puntos De Trayectoria//
    Sino
        TiempoTray ← TiempoMovL
    Fin_Si

    Si TiempoTray ← 0 Entonces
        VelT ← 0
    Sino
        VelT ← Mag_U / TiempoTray //Velocidad Entre Puntos De Trayectoria//
    Fin_Si

//Rutina Para Calcular La Velocidad Lineal En Cada Punto De La Trayectoria//
    Si i < Fila Entonces
        Si Mag_U = 0 Entonces
            VelX ← 0
            VelY ← 0
            VelZ ← 0
        Sino
            VelX ← (Trayectoria(i + 1, 1) - Trayectoria(i, 1)) / TiempoTray
            VelY ← (Trayectoria(i + 1, 2) - Trayectoria(i, 2)) / TiempoTray
            VelZ ← (Trayectoria(i + 1, 3) - Trayectoria(i, 3)) / TiempoTray
        Fin_Si
    Sino
        VelX ← VelX
        VelY ← VelY
        VelZ ← VelZ
    Fin_Si

//Llama Función Para Cálculo De Criterios De Optimización Manipulabilidad Y Consumo
Eléctrico En Punto Dado De La Trayectoria Y Almacena Resultado En Vector Punto_Cr()//
    Punto_Cr ← Criterios_optL(Vect_CInversa(),d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3,
d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ, AcWX,

```

AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(),
Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)

//Rutina Para Almacenar Valores Hallados Con Función De Cálculo De Criterios En Matriz
De Salida//

```
Para j ← 2 Hasta 64 Hacer
    Mat_Salida(i + 1, j) ← Punto_Cr(1, j)
Next j
```

//Rutina Para Almacenar Magnitudes, Velocidades Y Tiempos Corregidos Entre Puntos De
La Trayectoria En Matriz De Salida //

```
Mat_Salida(i + 1, 61) ← Mag_U
Mat_Salida(i + 1, 62) ← VelT
Mat_Salida(i + 1, 63) ← TiempoTray
```

//Almacena El Número Del Punto De Trayectoria En Matriz De Salida//

```
Mat_Salida(i + 1, 1) ← i + 1 - 1
```

//Calcula El Consumo En KWH Entre Puntos De La Trayectoria Y Lo Almacena En La Matriz
De Salida//

```
Mat_Salida(i + 1, 70) ← Mat_Salida(i + 1, 56) * TiempoTray / 3600
```

//Sumatorias De Comprobación De Variables Para Definir Restricciones En Algoritmo De
Optimización//

//Sumatoria Comprobación Articulaciones//

```
sumaCTh1 ← sumaCTh1 + Punto_Cr(1, 14)
sumaCTh2 ← sumaCTh2 + Punto_Cr(1, 15)
sumaCTh3 ← sumaCTh3 + Punto_Cr(1, 16)
sumaCTh4 ← sumaCTh4 + Punto_Cr(1, 17)
sumaCTh5 ← sumaCTh5 + Punto_Cr(1, 18)
sumaCTh6 ← sumaCTh6 + Punto_Cr(1, 19)
```

//Sumatoria Comprobación Velocidades//

```
sumaCdTh1 ← sumaCdTh1 + Punto_Cr(1, 26)
sumaCdTh2 ← sumaCdTh2 + Punto_Cr(1, 27)
sumaCdTh3 ← sumaCdTh3 + Punto_Cr(1, 28)
sumaCdTh4 ← sumaCdTh4 + Punto_Cr(1, 29)
sumaCdTh5 ← sumaCdTh5 + Punto_Cr(1, 30)
sumaCdTh6 ← sumaCdTh6 + Punto_Cr(1, 31)
```

//Sumatoria Comprobación Potencia//

```
sumaCPot ← sumaCPot + Punto_Cr(1, 57)
```

//Sumatoria 1-Manipulabilidad//

```
suma1_w ← suma1_w + Punto_Cr(1, 60)
```

//Sumatoria Tiempo trayectoria//

```
sumaTiempo ← sumaTiempo + TiempoTray
```

//Sumatoria Consumo en KWH//

```
SumaCons ← SumaCons + Mat_Salida(i + 1, 64)
```

Fin_Para

```

//
//Calcula 1-Manipulabilidad Promedio Trayectoria//
suma1_w ← suma1_w / (Fila)

//Almacena Sumatorias Para Definir Restricciones De Algoritmo Optimización En Matriz De Salida//
Mat_Salida(Fila + 2, 14) ← sumaCTh1
Mat_Salida(Fila + 2, 15) ← sumaCTh2
Mat_Salida(Fila + 2, 16) ← sumaCTh3
Mat_Salida(Fila + 2, 17) ← sumaCTh4
Mat_Salida(Fila + 2, 18) ← sumaCTh5
Mat_Salida(Fila + 2, 19) ← sumaCTh6
Mat_Salida(Fila + 2, 26) ← sumaCdTh1
Mat_Salida(Fila + 2, 27) ← sumaCdTh2
Mat_Salida(Fila + 2, 28) ← sumaCdTh3
Mat_Salida(Fila + 2, 29) ← sumaCdTh4
Mat_Salida(Fila + 2, 30) ← sumaCdTh5
Mat_Salida(Fila + 2, 31) ← sumaCdTh6
Mat_Salida(Fila + 2, 57) ← sumaCPot
Mat_Salida(Fila + 2, 60) ← suma1_w
Mat_Salida(Fila + 2, 63) ← sumaTiempo
Mat_Salida(Fila + 2, 64) ← SumaCons
//
Fin Función Valid_Trayectoria

```

Function **Opt_HKA**(Trayectoria(), Vect_Espacio(), POpt_Consumo, d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ, AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(), Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn, min_1_W) Devolver (T_{Optim}())

```
//FUNCION OPTIMIZACION DE TRAYECTORIAS HKA//
```

```

//Número De Variables q (N_q)//
// q1: Radio espacio de trabajo (U_Radio)//
//q2: Angulo de colatitud (U_AngC)
// q3: Angulo azimutal (U_AngA)//
N_q ← 3

//Valores Vector Medias//
U_Radio ← (Vect_Espacio(2, 1) + Vect_Espacio(1, 1)) / 2
U_AngC ← (Vect_Espacio(4, 1) + Vect_Espacio(3, 1)) / 2
U_AngA ← (Vect_Espacio(6, 1) + Vect_Espacio(5, 1)) / 2

//Valores Diagonal Matriz Covari_1anza//
Var_Radio ← ((Vect_Espacio(2, 1) - Vect_Espacio(1, 1)) / 6) ^ 2
Var_AngC ← ((Vect_Espacio(4, 1) - Vect_Espacio(3, 1)) / 6) ^ 2

```

```

Var_AngA ← ((Vect_Espacio(6, 1) - Vect_Espacio(5, 1)) / 6) ^ 2

//Número De Muestras (O Trayectorias)//
N_m ← 50

//Número De Mejores Candidatos//
N_E ← 5

//Coeficiente De Ralentización//
Coef ← 0.9

//Factor De Peso Para Resaltar Restricciones//
FactPeso ← 100

//Cantidad Máxima De Iteraciones De Algoritmo//
iter_K ← 200

//Media Iteración 0//
Mediai_1(1, 1) ← U_Radio
Mediai_1(1, 2) ← U_AngC
Mediai_1(1, 3) ← U_AngA

//Matriz Varianza-Covari_1anza Iteración 0//
Para i ← 1 Hasta 3 Hacer
    Para j ← 1 Hasta 3 Hacer
        Covari_1(i, j) ← 0
    Fin_Para
Fin_Para
Covari_1(1, 1) ← Var_Radio
Covari_1(2, 2) ← Var_AngC
Covari_1(3, 3) ← Var_AngA

Fila ← LimiteSuperior (Trayectoria, 1)

//Llama Función Calculo y Validación Criterios Optimización Consumo Eléctrico y
Manipulabilidad Trayectoria Original//
Tray_Original ← Valid_Trayectoria(Trayectoria(), D_px, D_py, D_pzd_1, a_1, alfa_1, a_2,
d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX,
AcY, AcZ, AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(),
IPist(), b_(), Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)

Manip_inic ← Tray_Original(Fila + 2, 60) //1-w Promedio Trayectoria Original//

//Encabezados Matriz Salida HKA//
HKA(1, 1) ← "K"
HKA(3, 1) ← "u"
HKA(4, 1) ← "s"
HKA(5, 1) ← "J(q)"

```

```

HKA(6, 1) ← "Cons(KWH)"
HKA(7, 1) ← "(1-w)"
HKA(8, 1) ← "Tpo(seg)"

C_Paro ← 0 //Indicador de condición de paro//
//
Mientras (k_ <= iter_K) Y (C_Paro < 10) Hacer
//RUTINA CALCULO ALGORITMO HKA EN ITERACIÓN K//
//
    Para i ← 1 Hasta N_m Hacer
    //Rutina para calcular N_m Trayectorias con parámetros algoritmos iteración k//
    //
    //Mat_q: Matriz Muestras. Almacena valores variables decisión, restricciones, y función de
    costo J(q) de las N_m Trayectorias//
        Mat_q(i + 1, 1) ← i //Almacena Número De Muestra en Matriz Muestras//

        Si i = N_m Entonces
        //En última iteración se halla J(q) con Media uncial//
            Vec_q ← Mediai_1 // Vec_q : Vector Variables Decisión.
        Sino
            Vec_q ← MNormRand(Mediai_1(), Covari_1())
        Fin_Si

        Para j ← 1 Hasta N_q Hacer
        //Rutina Para Almacenar Valores Vector Variables De Decisión En Matriz Muestras//
            Mat_q(i + 1, j + 1) ← Vec_q (1, j) //
        Fin_Para

    //Cálculo de Deltas A Adicionar En Trayectoria//
        D_px ← Cos(Vec_q (1, 3)) * Abs(Cos(Vec_q (1, 2)) * Vec_q (1, 1)) - Trayectoria(1, 1)
        D_py ← Sen(Vec_q (1, 3)) * Abs(Cos(Vec_q (1, 2)) * Vec_q (1, 1)) - Trayectoria(1, 2)
        D_pz ← -Sen(Vec_q (1, 2)) * Vec_q (1, 1) + d_1 - Trayectoria(1, 3)

        Tray_Valid ← Valid_Trayectoria(Trayectoria(),D_px, D_py, D_pzd_1, a_1, alfa_1,
        a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY,
        WZ, AcX, AcY, AcZ, AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(),
        In_5(), In_6(), IPist(), b_(), Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)

    //Calculo restricciones//
    //Sumatoria Comprobación Articulaciones Por Factor de Peso y Se Almacena en Matriz
    Muestras//
        Mat_q(i + 1, 5) ← Tray_Valid(Fila + 2, 14) * FactPeso //g1(q)//
        Mat_q(i + 1, 6) ← Tray_Valid(Fila + 2, 15) * FactPeso //g2(q)//
        Mat_q(i + 1, 7) ← Tray_Valid(Fila + 2, 16) * FactPeso //g3(q)//
        Mat_q(i + 1, 8) ← Tray_Valid(Fila + 2, 17) * FactPeso //g4(q)//
        Mat_q(i + 1, 9) ← Tray_Valid(Fila + 2, 18) * FactPeso //g5(q)//
        Mat_q(i + 1, 10) ← Tray_Valid(Fila + 2, 19) * FactPeso //g6(q)//

```

```
//Sumatoria Comprobación Velocidades Por Factor de Peso y Se Almacena en Matriz
Muestras //
```

```
Mat_q(i + 1, 11) ← Tray_Valid(Fila + 2, 26) * FactPeso //g7(q)//
Mat_q(i + 1, 12) ← Tray_Valid(Fila + 2, 27) * FactPeso //g8(q)//
Mat_q(i + 1, 13) ← Tray_Valid(Fila + 2, 28) * FactPeso //g9(q)//
Mat_q(i + 1, 14) ← Tray_Valid(Fila + 2, 29) * FactPeso //g10(q)//
Mat_q(i + 1, 15) ← Tray_Valid(Fila + 2, 30) * FactPeso //g11(q)//
Mat_q(i + 1, 16) ← Tray_Valid(Fila + 2, 31) * FactPeso //g12(q)//
```

```
//Sumatoria Comprobación Potencia Por Factor de Peso y Se Almacena en Matriz
Muestras//
```

```
Mat_q(i + 1, 17) ← Tray_Valid(Fila + 2, 57) * FactPeso //g13(q)//
```

```
//J(q) = SumaPot * POpt_Consumo+suma1_w * (1 - POpt_Consumo) Peso y Se Almacena
en Matriz Muestras//
```

```
Mat_q(i + 1, 18) ← (Tray_Valid(Fila + 2, 64) / (Cons_nom / 3600
* Tray_Valid(Fila + 2, 63))) * POpt_Consumo + (Tray_Valid(Fila + 2, 60) - min_1_W)
/ (1 - min_1_W) * (1 - POpt_Consumo)
```

```
//Sumatoria Restricciones Peso y Se Almacena en Matriz Muestras//
```

```
Mat_q(i + 1, 19) ← Mat_q(i + 1, 5) + Mat_q(i + 1, 6) + Mat_q(i + 1, 7)
+ Mat_q(i + 1, 8) + Mat_q(i + 1, 9) + Mat_q(i + 1, 10) + Mat_q(i + 1, 11)
+ Mat_q(i + 1, 12) + Mat_q(i + 1, 13) + Mat_q(i + 1, 14) + Mat_q(i + 1, 15)
+ Mat_q(i + 1, 16) + Mat_q(i + 1, 17)
```

```
//Jnew(q) = J(q)+ Sumatoria Restricciones Peso y Se Almacena en Matriz Muestras//
```

```
Mat_q(i + 1, 20) ← Mat_q(i + 1, 18) + Mat_q(i + 1, 19)
```

```
Mat_q(i + 1, 21) ← Tray_Valid(Fila + 2, 64) //Consumo total en KWH trayectoria//
```

```
Mat_q(i + 1, 22) ← Tray_Valid(Fila + 2, 60) //(1-w) Promedio//
```

```
Mat_q(i + 1, 23) ← Tray_Valid(Fila + 2, 63) //Tiempo total trayectoria en seg//
```

```
//
```

```
Fin_Para
```

```
Para i ← 1 Hasta N_q Hacer
```

```
SumaN_E(1, i) ← 0 //Vector Sumatoria N_E mejores candidatos qi//
```

```
Suma_qk_ek_2(1, i) ← 0 //Vector Sumatoria cuadrada de qi-medición (qi-E_k)^2//
```

```
Fin_Para
```

```
Para i ← 1 Hasta N_E Hacer
```

```
//Rutina Para Ordenar Matriz Muestras Según Jnew(q) De Menor A Mayor//
```

```
//
```

```
Para j ← (i + 1) Hasta N_m - 1 Hacer
```

```
Si Mat_q(i + 1, 20) > Mat_q(j + 1, 20) Entonces
```

```
Para m_ ← 1 Hasta 22
```

```
temp ← Mat_q(i + 1, m_)
```

```
Mat_q(i + 1, m_) ← Mat_q(j + 1, m_)
```

```
Mat_q(j + 1, m_) ← temp
```

```

        Fin_Para
        Fin_Si
    Fin_Para
    Para n_ ← 1 Hasta N_q Hacer
        //Vector Sumatoria N_E mejores candidatos qi//
        SumaN_E(1, n_) ← SumaN_E(1, n_) + Mat_q(i + 1, n_ + 1)
    Fin_Para
//
Fin_Para

Para i ← 1 To N_q
    //Vector Medición de la iteración k//
    E_k(1, i) ← 1 / N_E * SumaN_E(1, i)
Fin_Para

Para i ← 1 To N_E
    //Vector Sumatoria cuadrada de qi-medición (qi-E_k)^2//
    Para n_ ← 1 To N_q
        Suma_qk_ek_2(1, n_) ← Suma_qk_ek_2(1, n_) + (Mat_q(i + 1, n_ + 1)
            - E_k(1, n_)) ^ 2
    Fin_Para
Fin_Para

Suma_sqr_Vk ← 0 //Sumatoria raiz V_k//
El_max ← 0 //Inicialización para almacenar máximo V_k//

//Almacenar # Iteración En Matriz Salida HKA//
HKA(1, k_ * N_q + 2) ← k_

Para i ← 1 Hasta N_q Hacer
//
    //Vector diagonal de Matriz de Varianza//
    V_k(1, i) ← 1 / N_E * Suma_qk_ek_2(1, i)

    //Vector A Determinar Para Asegurar Una Óptima Estimación//
    L_k(1, i) ← Covari_1(i, i) / (Covari_1(i, i) + V_k(1, i))

    //Valor Índice q y Almacena En Matriz Salida HKA
    HKA(2, k_ * N_q + i + 1) ← i

    //Almacena Valores Matriz Media En Matriz De Salida HKA
    HKA(3, k_ * N_q + i + 1) ← Mediai_1(1, i)

    //Error De Estimación Previo En Iteración k, Igual A Matriz Media k+1//
    Mediai_1(1, i) ← Mediai_1(1, i) + L_k(1, i) * (E_k(1, i) - Mediai_1(1, i))

    //Vector Raíz V_k//
    sqr_Vk(1, i) ← Raíz(V_k(1, i))

```

```

//Sumatoria Raíz V_k//
Suma_sqr_Vk ← Suma_sqr_Vk + sqr_Vk(1, i)

Si El_max < V_k(1, i) Entonces El_max ← V_k(1, i)
//
Fin_Para

//Valores Para Calcular Factor De Ralentización//
El_min ← (1 / N_q * Suma_sqr_Vk) ^ 2
Si El_min > 1 Entonces El_min ← 1

//Factor De Ralentización//
ak_ ← Coef * El_min / (El_min + El_max)

//Matriz De Varianza-Covari_1anza Previa//
Para i ← 1 Hasta N_q Hacer
//Rutina para almacenar Valores Iniciales Matriz Varianza-Covarianza En Matriz De Salida
HKA//
HKA(4, k_ * N_q + i + 1) ← Covari_1(i, i)
Covari_1(i, i) ← (1 - ak_ * L_k(1, i)) * Covari_1(i, i)
Fin_Para

Para m_ ← 1 Hasta 23 Hacer
//Ubica Jnew(q) hallado con Mediai_1 en posición 6 (N_m=6)//
Mat_q(7, m_) ← Mat_q(N_m + 1, m_)
Fin_para

//Guarda En Matriz De Salida HKA Valor De J(q), Consumo, 1-w y Tiempo Obtenido Con
Mediai_1//
HKA(5, k_ * N_q + 2) ← Mat_q(7, 20) 'J(x)
HKA(6, k_ * N_q + 2) ← Mat_q(7, 21) 'Consumo (KWH)
HKA(7, k_ * N_q + 2) ← Mat_q(7, 22) '(1-w)
HKA(8, k_ * N_q + 2) ← Mat_q(7, 23) 'Tiempo

//CHEQUEO CONDICION DE PARO//
Si k_ > 9 Entonces
Si (Abs(HKA(7, k_ * N_q + 2) - HKA(7, (k_ - 1) * N_q + 2)) < 0.0005)
And (HKA(5, k_ * N_q + 2) < 0.9) And (Abs(HKA(6, k_ * N_q + 2)
- HKA(6, (k_ - 1) * N_q + 2)) < 0.0000001) And (Abs(HKA(6, k_ * N_q + 2) /
(Cons_nom / 3600 * HKA(8, k_ * N_q + 2)) - HKA(6, (k_ - 1) * N_q + 2) /
(Cons_nom / 3600 * HKA(8, (k_ - 1) * N_q + 2))) < 0.0005) Entonces
C_Paro ← C_Paro + 1
Sino
C_Paro ← 0
Fin_Si
Fin_Si

```

```

    k_ ← k_ + 1
//
Fin_Mientras

Mat_q(1, 1) ← "NP"
Mat_q(1, 2) ← "Radio(x1)"
Mat_q(1, 3) ← "AngP(x2)"
Mat_q(1, 4) ← "AngR(x3)"
Mat_q(1, 5) ← "g1(x)"
Mat_q(1, 6) ← "g2(x)"
Mat_q(1, 7) ← "g3(x)"
Mat_q(1, 8) ← "g4(x)"
Mat_q(1, 9) ← "g5(x)"
Mat_q(1, 10) ← "g6(x)"
Mat_q(1, 11) ← "g7(x)"
Mat_q(1, 12) ← "g8(x)"
Mat_q(1, 13) ← "g9(x)"
Mat_q(1, 14) ← "g10(x)"
Mat_q(1, 15) ← "g11(x)"
Mat_q(1, 16) ← "g12(x)"
Mat_q(1, 17) ← "g13(x)"
Mat_q(1, 18) ← "J(x)"
Mat_q(1, 19) ← "max(gi(x),0)"
Mat_q(1, 20) ← "Jnew(x)"
Mat_q(1, 21) ← "POT"
Mat_q(1, 22) ← "(1-W)"
Mat_q(1, 23) ← "Tpo(seg)"

//Deltas A Adicionar En Trayectoria//
D_px ← Cos(Mediai_1(1, 3)) * Abs(Cos(Mediai_1(1, 2)) * Mediai_1(1, 1)) - Trayectoria(1, 1)
D_py ← Sen(Mediai_1(1, 3)) * Abs(Cos(Mediai_1(1, 2)) * Mediai_1(1, 1)) - Trayectoria(1, 2)
D_pz ← -Sen(Mediai_1(1, 2)) * Mediai_1(1, 1) + d_1 - Trayectoria(1, 3)

T_optim ← Valid_Trayectoria(Trayectoria(), D_px, D_py, D_pzd_1, a_1, alfa_1, a_2, d_3, a_3,
alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ,
AcWX, AcWY, AcWZ, M_s(), M_m(), ln_1(), ln_2(), ln_3(), ln_4(), ln_5(), ln_6(), IPist(), b_(),
Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn)
//
Fin Función Opt_HKA

```

ALGORITMO PSEUDOCÓDIGO INTEGRADOR DE FUNCIONES

```

//CONSTANTES//
Pi1 = 3.14159265358979

//VARIABLES//
Texto : ComboRobot, ComboTray

```

Real: POpt_Consumo

```
//CODIGO PRINCIPAL//
```

```
Inicio
```

```
//SECCION CARGA DE PARAMETROS ROBOT//
```

```
  Escribir "Seleccione Robot"
```

```
  Leer ComboRobot
```

```
  N_Robot ← ComboRobot & ".xlsx"
```

```
  //Lee Archivo Excel de parámetros de robot y lo asigna a xlBook//
```

```
  Set xlBook ← xlApp.Workbooks.Open(N_Robot)
```

```
  //Parámetros Cinemática Directa//
```

```
  d_1 ← xlBook.Worksheets("Parametros").Cells(11, 2)
```

```
  a_1 ← xlBook.Worksheets("Parametros").Cells(11, 3)
```

```
  alfa_1 ← xlBook.Worksheets("Parametros").Cells(11, 4) * Pi1 / 180
```

```
  a_2 ← xlBook.Worksheets("Parametros").Cells(12, 3)
```

```
  d_3 ← xlBook.Worksheets("Parametros").Cells(13, 2)
```

```
  a_3 ← xlBook.Worksheets("Parametros").Cells(13, 3)
```

```
  alfa_3 ← xlBook.Worksheets("Parametros").Cells(13, 4) * Pi1 / 180
```

```
  d_4 ← xlBook.Worksheets("Parametros").Cells(14, 2)
```

```
  alfa_4 ← xlBook.Worksheets("Parametros").Cells(14, 4) * Pi1 / 180
```

```
  alfa_5 ← xlBook.Worksheets("Parametros").Cells(15, 4) * Pi1 / 180
```

```
  d_6 ← xlBook.Worksheets("Parametros").Cells(16, 2)
```

```
  d_h ← xlBook.Worksheets("Parametros").Cells(127, 3)
```

```
  a_h ← xlBook.Worksheets("Parametros").Cells(128, 3)
```

```
  //Matriz Límite Articulaciones Robot (rad) //
```

```
  //Límites Inferiores//
```

```
  Para i ← 1 Hasta 6 Hacer
```

```
    L_ART(i, 1) ← xlBook.Worksheets("Parametros").Cells(19 + i, 2) * Pi1 / 180
```

```
  Fin_Para
```

```
  //Límites Superiores//
```

```
  Para i ← 1 Hasta 6 Hacer
```

```
    L_ART(i, 2) ← xlBook.Worksheets("Parametros").Cells(19 + i, 3) * Pi1 / 180
```

```
  Fin_Para
```

```
  //Vector Límite Velocidades Articulaciones Robot (rad/seg) //
```

```
  Para i ← 1 Hasta 6 Hacer
```

```
    L_VART (i, 1) ← xlBook.Worksheets("Parametros").Cells(28 + i, 2) * Pi1 / 180
```

```
  Fin_Para
```

```
  //Velocidad Lineal Máxima Soldadura Efecto Final (m/seg) Por Catálogo//
```

```
  VelS ← xlBook.Worksheets("Parametros").Cells(3, 40)
```

```
  //Velocidades Efecto Final (Pistola)//
```

```
  //Velocidades Angular En rad/seg//
```

```
  WX ← xlBook.Worksheets("Parametros").Cells(42, 2) * Pi1 / 180
```

```
  WY ← xlBook.Worksheets("Parametros").Cells(43, 2) * Pi1 / 180
```

```

WZ ← xlBook.Worksheets("Parametros").Cells(44, 2) * Pi1 / 180

//Aceleraciones Efector Final (Pistola)//
//Aceleraciones Angular En rad/seg^2//
AcWX ← xlBook.Worksheets("Parametros").Cells(49, 2) * Pi1 / 180
AcWY ← xlBook.Worksheets("Parametros").Cells(50, 2) * Pi1 / 180
AcWZ ← xlBook.Worksheets("Parametros").Cells(51, 2) * Pi1 / 180
// Aceleraciones lineales en m/seg^2//
AcX ← xlBook.Worksheets("Parametros").Cells(52, 2)
AcY ← xlBook.Worksheets("Parametros").Cells(53, 2)
AcZ ← xlBook.Worksheets("Parametros").Cells(54, 2)

//Parámetros Dinámicos//
//Matriz Centro De Masa Eslabones Robot (m)//
Para i ← 1 Hasta 6 Hacer
    Para j ← 1 Hasta 3 Hacer
        M_s(i, j) ← xlBook.Worksheets("Parametros").Cells(60 + i, 1+j)
    Fin_Para
Fin_Para
//Valores Centro De Masa Pistola Robot (m)//
M_s(7, 1) ← xlBook.Worksheets("Parametros").Cells(133, 1)
M_s(7, 2) ← xlBook.Worksheets("Parametros").Cells(133, 2)
M_s(7, 3) ← xlBook.Worksheets("Parametros").Cells(133, 3)

//Matrices Momento Inercia Eslabones Robot (Kg.m^2)//
Para i ← 1 Hasta 3 Hacer
    Para j ← 1 Hasta 3 Hacer
        In_1(i, j) ← xlBook.Worksheets("Parametros").Cells(77 + i, j) //Inercia Eslabón 1//
        In_2(i, j) ← xlBook.Worksheets("Parametros").Cells(81 + i, j) //Inercia Eslabón 2//
        In_3(i, j) ← xlBook.Worksheets("Parametros").Cells(85 + i, j) //Inercia Eslabón 3//
        In_4(i, j) ← xlBook.Worksheets("Parametros").Cells(89 + i, j) //Inercia Eslabón 4//
        In_5(i, j) ← xlBook.Worksheets("Parametros").Cells(93 + i, j) //Inercia Eslabón 5//
        In_6(i, j) ← xlBook.Worksheets("Parametros").Cells(97 + i, j) //Inercia Eslabón 6//
        IPist(i, j) ← xlBook.Worksheets("Parametros").Cells(135 + i, j) //Inercia Pistola//
    Fin_Para
Fin_Para

Para i ← 1 Hasta 6 Hacer
    //Coeficientes Viscosidad Motores Robot (N.M.S/Rad)//
    b_(i, 1) ← xlBook.Worksheets("Parametros").Cells(104, j)
    //Constante Torque Motores Robot(N.m/A)//
    Kt(i, 1) ← xlBook.Worksheets("Parametros").Cells(108, j)
    //Constante De Fuerza Contra Electromotriz Motores Robot(V.s/rad)//
    Ke(i, 1) xlBook.Worksheets("Parametros").Cells(112, j)
    //Resistencias Armaduras Motores Robot(Ohms)//
    R_(i, 1) ← xlBook.Worksheets("Parametros").Cells(116, j)
    //Relación De Transmisión Motores Robot//
    nr(i, 1) ← xlBook.Worksheets("Parametros").Cells(120, j)

```

```

Fin_Para

//Potencia Nominal Robot (Kw)//
Pn ← xlBook.Worksheets("Parametros").Cells(122, 3)
//
//SECCION CARGA TRAYECTORIA ROBOT//
Escribir "Seleccione Trayectoria"
Leer ComboTray
Tray_Esc← ComboTray & ".xlsx"
//Lee Archivo Excel de Trayectorias y lo asigna a xlBook2//
Set xlBook2 ← xlApp.Workbooks.Open(Tray_Esc)

fin ← 1
//Chequea Número De Puntos De Trayectoria//
Mientras xlBook2.Worksheets(1).Cells(1 + fin, 1) <> "" Hacer
    fin ← 1 + fin
Fin_Mientras

Para i ← 1 Hasta (fin - 1) Hacer
    T_Analisis(i, 1) ← xlBook2.Worksheets(1).Cells(1 + i, 2)
    T_Analisis(i, 2) ← xlBook2.Worksheets(1).Cells(1 + i, 3)
    T_Analisis(i, 3) ← xlBook2.Worksheets(1).Cells(1 + i, 4)
    T_Analisis(i, 4) ← xlBook2.Worksheets(1).Cells(1 + i, 5)
    T_Analisis(i, 5) ← xlBook2.Worksheets(1).Cells(1 + i, 6)
    T_Analisis(i, 6) ← xlBook2.Worksheets(1).Cells(1 + i, 7)
Fin_Para

//
//SECCION CALCULO MANIPULABILIDAD MAXIMA MANIPULADOR//
wMax ← 0
Para i ← 0 Hasta 720 Con Paso 5 Hacer
    Theta2b ← i * Pi / 180
    Si i > 360 Entonces Theta2b ← (360 - i) * Pi / 180
    Para j ← i Hasta 720 Con Paso 5 Hacer
        Theta3b ← j * Pi / 180
        Si j > 360 Entonces Theta3b ← (360 - j) * Pi / 180
        //Manipulabilidad Brazo wA//
        wA ← Abs(a_2 * (d_4 * Cos(Theta3b) + a_3 * Sen (Theta3b))*(a_1 + a_3 *
            Cos(Theta2b + Theta3b) - d_4 * Sen(Theta2b + Theta3b) + a_2 Cos (Theta2b)))
        Si wA > wMax Entonces
            wMax ← wA
            Theta2bMax ← Theta2b
            Theta3bMax ← Theta3b
        Fin_Si
    Fin_Para
Fin_Para

min_1_W ← 1 - wMax
//

```

```

//SECCION DE CALCULO CON FUNCIONES CREADAS
//Cálculo Límites Espacio Esférico Aproximado Robot y Almacenar En Vect_Espacio (6x1)//
Vect_Espacio ← Espacio_Trab(d_1, a_1, alfa_1, a_2, d_3, a_3, alfa_3, d_4, alfa_4, alfa_5,
d_6, d_h, a_h, L_ART())

//Se define Peso Para Consumo en Algoritmo HKA//
Escribir "Introduzca Peso Para Consumo (entre 0 y 1)"
Leer POpt_Consumo

TOptim() ← Opt_HKA(TAnálisis()), Vect_Espacio (), POpt_Consumo, d_1, a_1, alfa_1, a_2, d_3,
a_3, alfa_3, d_4, alfa_4, alfa_5, d_6, d_h, a_h, VelX, VelY, VelZ, WX, WY, WZ, AcX, AcY, AcZ,
AcWX, AcWY, AcWZ, M_s(), M_m(), In_1(), In_2(), In_3(), In_4(), In_5(), In_6(), IPist(), b_(),
Kt(), Ke(), R_(), nr(), L_ART(), L_VART(), Pn, min_1_W

//
Fin

```