General Lagrangian Jacobian motion planning algorithm for affine robotic systems with application to a space manipulator

Krzysztof Tchoń and Joanna Ratajczak
Chair of Cybernetics and Robotics
Electronics Faculty, Wrocław University of Science and Technology
Wrocław, Poland
Email: krzysztof.tchon|joanna.ratajczak@pwr.edu.pl

***Abstract*—This paper proposes an extension of the concept of the General Lagrangian Jacobian Inverse from driftless to control affine robotic systems, and presents the corresponding Jacobian motion planning algorithm in the parametric form. A specific choice of the Lagrangian is recommended. The motion planning algorithm is applied to the motion planning problem of a free-floating space manipulator with non-zero momentum. A conjecture is formulated that for this specific choice of the Lagrangian the motion planning algorithm outperforms the other Jacobian algorithms in terms of the length of the resulting robot trajectory.**

I. Introduction

We shall study non-holonomic mobile robots and mobile manipulators from the perspective of control theory. From the control theoretic point of view a mobile robot could be represented either as a driftless or as a control affine system with output. The former representation refers to the kinematics of a robot subordinated to the Pfaffian motion constraints; the latter appears when the robot's dynamics are taken into account, either as a consequence of the dynamic reduction based on the d'Alembert Principle or of the momentum conservation. It follows that a control affine system describes the dynamics of wheeled mobile robots moving without the slip of their wheels or of the free-floating space robots conserving a non-zero momentum.

To both these control system representations a systematic methodology called the Endogenous Configuration Space Approach applies whose recent overview may be found in [1]. Its crucial ingredient is the end-point map defining the system's output (or state) at the end of the control time horizon, resulting from an application of a given control function. This map transforms the control space, referred to as the Endogenous Configuration Space, to the system's output. By differentiating the end-point map with respect to the control function we arrive at a concept of the system's Jacobian, and define the regular and singular endogenous configurations of the system. In this context the motion planning problem consists in finding a control function that drives the output at a prescribed time instant to a prescribed point. Solving the motion planning problem requires inverting the end-point map. This inversion can be based on the Jacobian, giving rise to the Jacobian motion planning algorithms. It has been shown that the Jacobian algorithms can be derived by reference to an optimal control problem formulated in the linear approximation of the (driftless or control affine) control system. If this problem employs the objective function in the general Lagrange form, the inverse is called the General Lagrangian Jacobian Inverse (in short GLJI), and the accompanied motion planning algorithm the General Lagrangian Jacobian Motion Planning Algorithm (GLJMPA). Mathematical foundations for GLJI and GLJMPA for driftless robotic systems have been laid recently in [2]. Moreover, in that reference we have shown that a specific choice of the Lagrange objective functions defines a bound on the energy and the length of the resulting trajectory of the linear approximation to the control system. The corresponding Jacobian inverse is named GLJI(AB). We want to mention that a preliminary form of GLJI, referred to as the Lagrangian Jacobian Inverse (LJI), has been introduced and examined in [3].

In this paper we transfer the basic results concerned with GLJI from the driftless to the control affine robotic systems, and derive a parametric form of GLJMPA for the latter systems. As a tentative area of application of GLJMPA we propose the space robotics, specifically the design of motion planning modules for free-floating space manipulators that perform in Space various kinds of interception maneuvers, [4], [5]. For this reason, as a performance test of GLJMPA, we study a motion planning problem of a free-floating space manipulator built recently in the Space Research Centre of the Polish Academy of Sciences [6]. This study is made under assumption that the robot's linear and angular momenta are equal to a non-zero constant. Since this robot is interesting on its own, its motion equations have been derived in terms of the Lagrangian mechanics, conditions for the momentum conservation are stated explicitly, then the control affine system representation is derived, and its controllability proved. The GLJMPA has been applied for several forms of the Lagrange objective function, and their performance compared. Relying on the results of computations a conjecture is put forward stating that for the motion planning that employs GLJI(AB) generates the shortest robot's trajectories within the whole

978-1-5386-2402-9/17/$31.00 ©2017 IEEE

class of GLJI.

This paper is organized in the following way. The next section introduces basics of Endogenous Configuration Space Approach for control affine systems, culminating in a general form of the Jacobian motion planning algorithm. In section III we define GLJI and the associated GLJMPA, and finally deliver its parametric form, useful for computations. Section IV describes the space manipulator, provides its control system representation, and shows its controllability. Results of numeric computations are included into section V. Section VI contains conclusions.

II. Basic concepts

As proclaimed the Introduction, we shall analyse the motion planning problem in robotic systems represented by a control affine system with output, of the form

$$\begin{cases} \dot{x} = f(x) + g(x)u = f(x) + \sum_{i=1}^{m} g_i(x)u_i \\ y = k(x) = (k_1(x), \dots, k_r(x)), \end{cases} \tag{1}$$

where $x \in \mathsf{R}^n$, $u \in \mathsf{R}^m$, $y \in \mathsf{R}^r$ denote, respectively, the state, the control and the output variable, the vector field $f(x)$ is called a drift, $g_1(x), g_2(x), \dots, g_m(x)$ are control vector fields, and $k(x)$ is the output function. We assume that the system (1) is controlled over a control time horizon $T > 0$. The feasible control functions belong to the space $\mathcal{X} = L^2_m[0, T]$ of Lebesque square integrable time functions, equipped with the inner product $\langle u_1(\cdot), u_2(\cdot)\rangle = \int_0^T u_1^T(t)u_2(t)dt$, that makes it an infinite dimensional Hilbert space. This space will be called further on the endogenous configuration space of the system (1). Given a control function $u(\cdot) \in \mathcal{X}$, we let $x(t) = \varphi_{x_0,t}(u(\cdot))$ denote the state trajectory of (1), initialized at x_0. Our standing assumption is that $x(t)$ exists for every control $u(\cdot)$ and every time instant $t \in [0, T]$. Having computed $x(t)$ we define the end-point map $K_{x_0,T} : \mathcal{X} \longrightarrow \mathsf{R}^r$ of the system (1) as follows

$$K_{x_0,T}(u(\cdot)) = y(T) = k(x(T)) = k(\varphi_{x_0,T}(u(\cdot))). \tag{2}$$

By definition, the end-point map attributes to any control function a value of the output function at T.

With reference to the end-point map, the motion planning problem for the system (1) amounts to determining a control function $u_d(\cdot) \in \mathcal{X}$, such that at T the system's output takes on the desired value y_d, i.e. $K_{x_0,T}(u_d(\cdot)) = y_d$. A solution to the motion planning problem can be provided by the inverse end-point map that computes the control function for a prescribed output value. Such an inverse end-point map is not unique, nevertheless certain inverses can be computed numerically on the basis of the system's Jacobian. At a fixed control function the Jacobian $J_{x_0,T}(u(\cdot)) : \mathcal{X} \longrightarrow \mathsf{R}^r$ is a linear map defined by differentiation of the end-point map with respect to the control, i.e. $J_{x_0,T}(u(\cdot)) = \mathsf{D}K_{x_0,T}(u(\cdot))$. It can be shown [7] that, given a control $u(t)$, the Jacobian is computed by the linear approximation to the system (1) along this control and the corresponding trajectory $x(t)$. To this objective, we find the matrices

$$A(t) = \frac{\partial(f(x(t)) + g(x(t))u(t))}{\partial x}, \quad B(t) = g(x(t)),$$
$$C(t) = \frac{\partial k(x(t))}{\partial x}, \tag{3}$$

of the linear approximation, and obtain the Jacobian as $\eta(T) = C(T)\xi(T)$, where, for a given $v(\cdot) \in \mathcal{X}$, $\xi(t) = \mathsf{D}\varphi_{x_0,t}(u(\cdot))v(\cdot) \in \mathsf{R}^n$, $\xi(0) = 0$, denotes the trajectory of a linear, time-dependent control system

$$\begin{cases} \dot{\xi} = A(t)\xi + B(t)v, \\ \eta(t) = C(t)\xi, \end{cases} \tag{4}$$

with output $\eta \in \mathsf{R}^r$. In conclusion, the Jacobian

$$J_{x_0,T}(u(\cdot))v(\cdot) = \eta(T) = C(T)\xi(T) =$$
$$C(T)\int_0^T \Phi(T, t)B(t)v(t)dt, \tag{5}$$

where the fundamental matrix $\Phi(t, s)$ solves the evolution equation $\frac{\partial\Phi(t,s)}{\partial t} = A(t)\Phi(t, s)$, with the initial condition $\Phi(s, s) = I_n$. Now, instead of inverting the end-point map, we shall invert the Jacobian. Specifically, the right inverse of the Jacobian (5), denoted by $J^{\#}_{x_0,T}(u(\cdot)) : \mathsf{R}^n \longrightarrow \mathcal{X}$, will be used as a map such that $J_{x_0,T}(u(\cdot))J^{\#}_{x_0,T}(u(\cdot)) = I_r$. Having chosen a right Jacobian inverse, and invoking a reasoning originated from the continuation method, the value of an inverse end-point map corresponding to a prescribed output y_d can determined as the limit $u_d(t) = \lim_{\theta\to+\infty} u_\theta(t)$ of the trajectory $u_\theta(\cdot)$ of the dynamic system

$$u'_\theta(\cdot) = -\gamma J^{\#}_{x_0,T}(u_\theta(\cdot))(K_{x_0,T}(u_\theta(\cdot)) - y_d),\ u_{\theta=0}(t) = u_0(t), \tag{6}$$

$u'(\cdot)$ denoting the derivative with respect to θ. As a matter of fact, the dynamic system (6) defines a motion planning algorithm. Its specific form will depend on the choice of the Jacobian right inverse. In the next section we shall device the Jacobian inverse called Lagrangian.

III. General Lagrangian Jacobian inverse and motion planning algorithm

Assuming that a control function $u(\cdot)$ has been fixed, the right inverse of the Jacobian provides a solution to the Jacobian equation

$$J_{x_0,T}(u(\cdot))v(\cdot) = \eta, \tag{7}$$

so that $v(\cdot) = J^{\#}_{x_0,T}(u(\cdot))\eta$. For the reason that the Jacobian acts on the infinite dimensional space of control functions, the solution to the Jacobian equation is by no means unique. To choose one of them, one can attach the Jacobian equation as an equality constraint to an optimal control problem addressed in the control system (4). The inverse produced as the solution to the optimal control problem with the Lagrange-type objective function and with the constraint (7) is called the General Lagrangian Jacobian Inverse (GLJI), see [2]. In formal terms we need to solve in the system (4) a quadratic optimal control problem

$$I = \min_{v(\cdot)} \frac{1}{2} \int_0^T \left(\xi^T(t), v^T(t)\right) \begin{bmatrix} Q(t) & S(t) \\ S^T(t) & R(t) \end{bmatrix} \begin{pmatrix} \xi(t) \\ v(t) \end{pmatrix} dt, \quad (8)$$

satisfying the equality constraint

$$J_{x_0,T}(u(\cdot))v(\cdot) = \eta. \quad (9)$$

The matrices in (8) should be chosen in such a way that $Q(t) = Q^T(t) \geq 0$, $R(t) = R^T(t) > 0$ and $Q(t) - S(t)R^{-1}(t)S^T(t) \geq 0$. Notice that, since the optimal control problem is solved for a fixed control function, the matrices $Q(t)$, $R(t)$ and $S(t)$ may depend on $u(\cdot) \in \mathcal{X}$. A complete derivation of GLJI based on the Pontryagin's Maximum Principle has been provided in [2]. If the matrix $S(t)$ in (8) is zero, we get the Lagrangian Jacobian Inverse (LJI) introduced in [3]. It is easily observed that after setting $Q(t) = 0$ and $S(t) = 0$, GLJI reduces to the well known Jacobian pseudoinverse (the Moore-Penrose's generalized inverse). A specific form of GLJI is obtained when we choose $Q(t) = A^T(t)A(t)$, $S(t) = A^T(t)B(t)$, and $R(t) = B^T(t)B(t)$, where $A(t)$ and $B(t)$ have appeared in (4), cf. [2]. In order to distinguish this form of GLJI further on it will be called GLJI(AB). It has been shown that in this case the length of the trajectory $\xi(t)$ is upper bounded as follows

$$\int_0^T ||\dot{\xi}(t)||dt \leq \sqrt{TI}. \quad (10)$$

As mentioned above, a general form of GLJI and of the corresponding motion planning algorithm (GLJMPA) in the infinite dimensional endogenous configuration space can be found in [2]. However, for computational purposes it is useful to perform a finite-dimensional parametrization of the control functions by expanding them into an orthogonal series and truncating the series after some number of terms. In this way a finite-dimensional right Jacobian inverse is obtained. For the driftless control system representation a relevant computational procedure has been provided in [8]. This procedure can be directly extended to control affine systems by merely taking the matrix $A(t)$ as defined in (3). We shall briefly summarize this procedure below.

We begin with choosing a truncated orthogonal basis $P_b(t) = [\varphi_0(t), \varphi_1(t), \ldots, \varphi_p(t)]$ in the space $\mathcal{X}$ of control functions. With respect to this basis any control function can be written down in the form

$$u_\lambda(t) = P(t)\lambda, \quad (11)$$

where $P(t)$ denotes an $m \times m(p+1)$-matrix composed of the basis functions, such that

$$P(t) = \text{blockdiag}\{P_b(t), P_b(t), \ldots, P_b(t)\}.$$

After the parametrization by λs, the control space becomes finite-dimensional, $\lambda \in \Lambda = \mathsf{R}^s$, for $s = m(p+1)$. Due to orthogonality $\int_0^T P^T(t)P(t)dt = I_s$. Next, given $u_\lambda(t)$, we compute the trajectory $x_\lambda(t)$ of the system (1), determine the planning error $e_\lambda = k(x_\lambda(T)) - y_d$, and find the matrices $A_\lambda(t)$, $B_\lambda(t)$, $C_\lambda(t)$ as well as $\Phi_\lambda(t,s)$, in compliance with (4). Continuing, for a certain $\mu \in \mathsf{R}^s$ and the corresponding $v_\mu(t) = P(t)\mu$ we compute $\xi_\lambda(t) = Z_\lambda(t)\mu$ where

$$Z_\lambda(t) = \int_0^t \Phi_\lambda(t,s)B_\lambda(s)P(s)ds.$$

Now, it is easily seen that the quadratic optimal control problem (8) with the constraint (9) becomes tantamount to the minimization of the quadratic objective function

$$\frac{1}{2}\mu^T W_\lambda \mu,$$

where

$$W_\lambda = \int_0^T \left[Z_\lambda^T(t)\ P^T(t)\right] \begin{bmatrix} Q(t) & S(t) \\ S^T(t) & R(t) \end{bmatrix} \begin{bmatrix} Z_\lambda(t) \\ P(t) \end{bmatrix} dt,$$

under the equality constraint $C_\lambda(T)Z_\lambda(T)\mu = \eta$. This problem can be solved by employing the technique of Lagrange multipliers, resulting in

$$\mu = W_\lambda^{-1}Z_\lambda^T(T)C_\lambda^T(T)\mathcal{M}_\lambda^{-1}\eta,$$

where $\mathcal{M}_\lambda = C_\lambda(T)Z_\lambda(T)W_\lambda^{-1}Z_\lambda^T(T)C_\lambda^T(T)$. Eventually, the parametric GLJMPA assumes the form

$$\lambda'(\theta) = -\gamma W_\lambda^{-1}Z_\lambda^T(T)C_\lambda^T(T)\mathcal{M}_\lambda^{-1}(k(x_\lambda(T)) - y_d). \quad (12)$$

Allowing θ to be discrete, $\theta = 0, 1, \ldots$, after an application to (12) of the Euler integration scheme we obtain a parametrized and discrete motion planning algorithm, that relies on starting from a λ_0 and sequentially updating the parameters λ in accordance with the following rule

$$\lambda_{\theta+1} = \lambda_\theta - \gamma W_{\lambda_\theta}^{-1}Z_{\lambda_\theta}^T(T)C_{\lambda_\theta}^T(T)\mathcal{M}_{\lambda_\theta}^{-1}(k(x_{\lambda_\theta}(T)) - y_d). \quad (13)$$

In the next section the parametrized discrete General Lagrangian Jacobian Motion Planning Algorithm (GLJMPA) will be applied to a motion planning problem of a free-floating space manipulator.

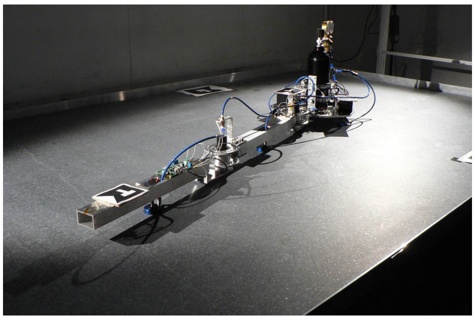

Fig. 1. SRC robot

IV. Space manipulator

As a test bed of performance of GLJMPA we shall use a free-floating space manipulator built in the Space Research Centre (SRC) of the Polish Academy of Sciences, described in [6] and presented in Figure 1. A schematic picture of SRC robot showing the robot's coordinates and parameters, is

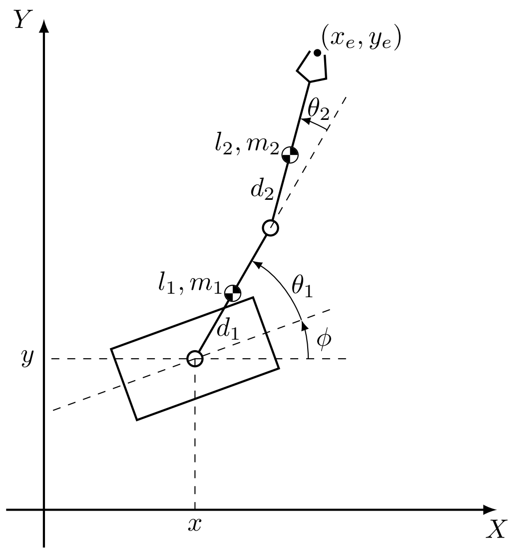

Fig. 2. SRC robot: schematic

displayed in Figure 2. As can be seen, SRC space manipulator consists of a mobile base (a satellite) and a 2DOF planar on-board manipulator. The robot is supported by air bearings that make it capable of floating horizontally over a granite table. The dynamic and geometric parameters of SRC robot are the following [4]: masses of the links $m_1 = 4.5kg$ and $m_2 = 1.5kg$, lengths of the links $l_1 = 0.619m$ and $l_2 = 0.6m$, positions of centers of mass on the links $d_1 = 0.313m$ and $d_2 = 0.287m$, the mass and the moment of inertia of the base $M = 12.9kg$ and $\mathrm{I} = 0.208kg{\cdot}m^2$. After choosing generalized coordinates $q = (x, y, \phi, \theta_1, \theta_2)^T \in \mathsf{R}^2 \times \mathsf{T}^3$, T^3 denoting the 3-dimensional torus, we find the Lagrangian

$$L(q,\dot q) = \frac{1}{2}(M{+}m_{12})(\dot x^2{+}\dot y^2){+}\frac{1}{2}\mathrm{I}\dot\phi^2{+}\frac{1}{2}(m_1 d_1^2{+}m_2 l_1^2)\times (\dot\phi + \dot\theta_1)^2 + \frac{1}{2}m_2 d_2^2(\dot\phi + \dot\theta_{12})^2 - (m_1 d_1 + m_2 l_1)(\dot\phi + \dot\theta_1)\times (\dot x s_{\phi+\theta_1} - \dot y c_{\phi+\theta_1}) - m_2 d_2(\dot\phi + \dot\theta_{12})(\dot x s_{\phi+\theta_{12}} - \dot y c_{\phi+\theta_{12}}) + m_2 l_1 d_2(\dot\phi + \dot\theta_1)(\dot\phi + \dot\theta_{12})c_{\theta_2}, \quad (14)$$

where $m_{12} = m_1 + m_2$, $\theta_{12} = \theta_1 + \theta_2$, and s_α, c_α denote $\sin\alpha$ and $\cos\alpha$. A glance at the Euler-Lagrange equations generated by the Lagrangian (14) reveals that, although the linear momenta are conserved, the angular momentum is not because the term $\frac{\partial L}{\partial \phi}$ does not vanish. However, it is well known [9] that the angular momentum is conserved, if instead of x, y we introduce the barycentric coordinates $(\bar x, \bar y)$ defined by the identities $(M + m_{12})\bar x = Mx + m_1(x + d_1 c_{\phi+\theta_1}) + m_2(x + l_1 c_{\phi+\theta_1} + d_2 c_{\phi+\theta_{12}})$ and $(M + m_{12})\bar y = My + m_1(y + d_1 s_{\phi+\theta_1}) + m_2(y + l_1 s_{\phi+\theta_1} + d_2 s_{\phi+\theta_{12}})$. A computation shows that now

$$L(\bar q,\dot{\bar q}) = \frac{1}{2}\mathrm{A}(\dot{\bar x}{+}\dot{\bar y}){+}\frac{1}{2}\mathrm{I}\dot\phi^2{+}\frac{1}{2}\mathrm{B}(\dot\phi{+}\dot\theta_1)^2{+}\frac{1}{2}\mathrm{C}(\dot\phi{+}\dot\theta_{12})^2 + \mathrm{D}c_{\theta_2}(\dot\phi + \dot\theta_1)(\dot\phi + \dot\theta_{12}), \quad (15)$$

where $\bar q = (\bar x, \bar y, \phi, \theta_1, \theta_2)^T$ and $\mathrm{A} = M + m_{12}$, $\mathrm{B} = \frac{m_1 m_2(l_1 - d_1)^2 + M(m_1 d_1^2 + m_2 l_1^2)}{M + m_{12}}$, $\mathrm{C} = \frac{(M+m_1)m_2 d_2^2}{M+m_{12}}$, $\mathrm{D} = \frac{m_1 m_2(l_1-d_1)d_2 + M m_2 l_1 d_2}{M+m_{12}}$. Now we have $\frac{\partial L}{\partial \phi} = 0$, so the angular momentum is conserved. Assuming that the linear momenta are equal to a_1 and a_2, and the angular momentum equals a_3, we get the uniform motion of the center of mass, $\mathrm{A}\dot{\bar x} = a_1$, $\mathrm{A}\dot{\bar y} = a_2$. In turn, the angular momentum conservation leads to the following identity

$$F(\theta_2)\dot\phi + G(\theta_2)\dot\theta_1 + H(\theta_2)\dot\theta_2 = a_3, \quad (16)$$

for $F(\theta_2) = \mathrm{I} + \mathrm{B} + \mathrm{C} + 2\mathrm{D}c_{\theta_2}$, $G(\theta_2) = \mathrm{B} + \mathrm{C} + 2\mathrm{D}c_{\theta_2}$, $H(\theta_2) = \mathrm{C} + \mathrm{D}c_{\theta_2}$. We shall assume that $a_3 \neq 0$. The conservation law (16) has an affine Pfaffian form $M(x)\dot x = a_3$, $x = (\phi, \theta_1, \theta_2)^T \in \mathsf{T}^3$. Therefore, it is easily checked that (16) can be equivalently expressed as a control affine system (1)

$$\dot x = M^{\#}(x)a_3 + W(x)u,$$

where $M^{\#}(x)$ denotes a right inverse of $M(x)$, and the columns of the matrix $W(x)$ span the null space of $M(x)$, so that $M(x)W(x) = 0$. After some calculations involving the pseudoinverse of $M(x)$ we arrive at the control affine system

$$\begin{cases} \dot\phi = \frac{F(\theta_2)a_3}{N(\theta_2)} - \frac{G(\theta_2)}{F(\theta_2)}u_1 - \frac{H(\theta_2)}{F(\theta_2)}u_2 \\ \dot\theta_1 = \frac{G(\theta_2)a_3}{N(\theta_2)} + u_1 \\ \dot\theta_2 = \frac{H(\theta_2)a_3}{N(\theta_2)} + u_2, \\ y = k(x) = (\phi, l_1 c_{\theta_1} + l_2 c_{\theta_{12}}, l_1 s_{\theta_1} + l_2 s_{\theta_{12}})^T, \end{cases} \quad (17)$$

where $N(\theta_2) = F^2(\theta_2) + G^2(\theta_2) + H^2(\theta_2)$, with the output function referring to the base orientation and the position of the end-effector in a base-fixed frame.

Differently to the driftless systems the controllability of a control affine system may be hard or even impossible to establish. Despite this, the case of (17) is relatively easy. To show that this system is controllable, we first apply a feedback $v_1 = \frac{G(\theta_2)a_3}{N(\theta_2)} + u_1$, $v_2 = \frac{H(\theta_2)a_3}{N(\theta_2)} + u_2$ that results in a controllable system

$$\begin{cases} \dot\phi = \frac{a_3}{F(\theta_2)} - \frac{G(\theta_2)}{F(\theta_2)}v_1 - \frac{H(\theta_2)}{F(\theta_2)}v_2 \\ \dot\theta_1 = v_1 \\ \dot\theta_2 = v_2. \end{cases} \quad (18)$$

Denote by $g_1(x)$ and $g_2(x)$ the control vector fields in (18), and by $f(x)$ the drift term. A computation of Lie brackets shows that outside the singular configurations $s_{\theta_2} = 0$ of the on board manipulator, the Lie algebra rank condition is satisfied by vector fields $g_1(x)$, $g_2(x)$, and $g_{12}(x) = [g_1, g_2](x) = (-\frac{2\mathrm{ID}}{F^2(\theta_2)}s_{\theta_2}, 0, 0)^T$. Taking instead of $g_{12}(x)$ the bracket $g_{212}(x) = [g_2, g_{12}](x) = (-2\mathrm{ID}\frac{F(\theta_2)c_{\theta_2} + 4Ds^2_{\theta_2}}{F^3(\theta_2)}, 0, 0)^T$ we notice that the Lie algebra rank condition is satisfied at any point of the torus T^3 by vector fields $g_1(x)$, $g_2(x)$, $g_{12}(x)$ and $g_{212}(x)$. The growth vector of the control distribution is either $(2, 3)$ or $(2, 2, 3)$. Having demonstrated the Lie algebra rank condition we discover that the drift vector field in (17) has zero divergence, $div f(x) = 0$, so it is Poisson stable. In this way, referring to an old theorem of Lobry, see [10] or

[11], we have established controllability of the control affine system (17). In the next section a motion planning problem for SRC space robot will be analysed, and solved using GLJMPA in the discrete parametric form (13). A number of specific GLJIs, designed in compliance with (8), will be employed to this objective.

V. Computations

As has been already written, the numerical experiments will be performed using various GLJIs. They have been specified in Table I. Additionally, for comparison proposes, we shall add the Jacobian Pseudoinverse (the Moore-Penrose's generalized inverse) that is also an instance of GLJI. By $J_4^{GLJI\#}$ we mean GLJI(AB) discussed in section III. The motion planning problem consists in finding a control function which drives the space manipulator (17) whose initial state is $x_0 = (0, 0, \pi/8)^T$ to the desired point $y_d = (0, 0.6, 0)^T$ in time $T = 20s$. The control function is represented by a truncated Fourier series, such that $P_b = (1, \sin(\omega t), \cos(\omega t), \sin(2\omega t), \cos(2\omega t))$, where $\omega = \frac{2\pi}{T}$. The initial value of control parameters is set to $\lambda_0 = (0, 0.04, 0, 0, 0, 0, 0.04, 0, 0, 0)^T$, the coefficient $a_3 = 0.02$ and the error decay rate $\gamma = 0.02$.

TABLE I
List of GLJIs

Notation	$Q(t)$	$S(t)$	$R(t)$
$J^{JP\#}$	$0_{3\times3}$	$0_{3\times2}$	I_2
$J^{LJI\#}$	I_3	$0_{3\times2}$	I_2
$J_1^{GLJI\#}$	I_3	$\begin{bmatrix} I_2 \\ 0_{1\times2} \end{bmatrix}$	I_2
$J_2^{GLJI\#}$	$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$	I_2
$J_3^{GLJI\#}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	I_2
$J_4^{GLJI\#}$	$A^T(t)A(t)$	$A^T(t)B(t)$	$B^T(t)B(t)$

Solutions of the motion planning problem are depicted in Figures 3 and 4. In addition, Figure 5 shows the Euclidean length $l_\theta = \int_0^T \|\dot{x}_\theta(t)\| dt$ of the resultant trajectory for each of the applied inverses. From Figures 3–6 one can learn that the choice of the $Q(t)$, $S(t)$ and $R(t)$ matrices has a significant influence on the solution, whereas the solution provided by GLJMPA involving $J_4^{GLJI\#}$ (GLJI(AB)) seems to be the most efficient in terms of the length of the state space trajectory. The corresponding end-effector trajectories $x_e = l_1 c_{\theta_1} + l_2 c_{\theta_{12}}$ and $y_e = l_1 s_{\theta_1} + l_2 s_{\theta_{12}}$ are shown in Figure 6.

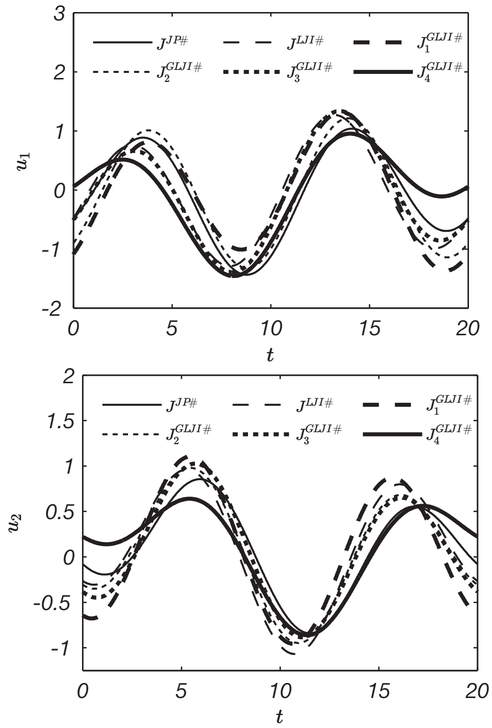

Fig. 3. SRC robot: controls

VI. Conclusion

This paper has extended the concept of General Lagrangian Jacobian Inverse (GLJI) from driftless to affine control systems, provided a parametric form of General Lagrangian Jacobian Motion Planning Algorithm (GLJMPA), and shown an application of this algorithm to the motion planning of a free-floating space robot. The results look promising, especially in the case of GLJI(AB). Relying on the results of section V it is conjectured that GLJI(AB) produces the shortest trajectories of the control affine robotic systems. An attempt at settling this conjecture will be a subject of our future studies.

Acknowledgment

This research was supported in part by the Wrocław University of Science and Technology under a statutory research project, and in part by the National Science Centre, Poland, under the grant No 2015/17/B/ST7/03995.

References

[1] K. Tchoń, Endogenous Configuration Space Approach: An Intersection of Robotics and Control Theory, in: *Nonlinear Systems*, Springer, 2017, pp. 209–234.
[2] K. Tchoń and J. Ratajczak, General Lagrange-type Jacobian inverse for non-holonomic robotic systems, submitted, 2017.
[3] K. Tchoń, A. Ratajczak and I. Góral, Lagrangian Jacobian inverse for nonholonomic robotic systems, *Nonlinear Dynamics*, 82, 1923–1932, 2015.
[4] T. Rybus, *Control of a satellite manipulator during the interception maneuver and the motion stabilization,* Ph.D. dissertation, Wrocław University of Science and Technology, 2016.
[5] T. Rybus, K. Seweryn and J. Sasiadek, Control System for Free-Floating Space Manipulator Based on Nonlinear Model Predictive Control (NMPC), *J. Intell. Robot. Syst.* 85, 491–509, 2017.
[6] T. Rybus and K. Seweryn, Planar air-bearing microgravity simulators: review of applications, existing solutions and design parameters, *Acta Astronaut.* 120, 239–259, 2016.
[7] E. D. Sontag, *Mathematical Control Theory*, Springer, 1985.
[8] K. Tchoń and I. Góral, Optimal motion planning for non-holonomic robotic systems, *Proc. IFAC World Congress*, 2017, to appear.

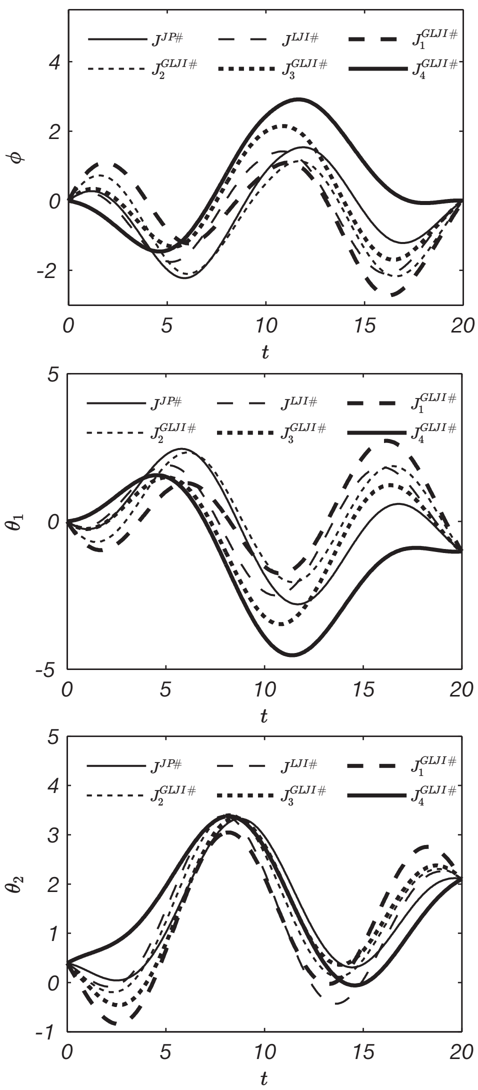

Fig. 4. SRC robot: state variable trajectories

[9] E. Papadopoulos, *On the dynamics and control of space manipulators*, Ph.D. dissertation, MIT, 1990.

[10] C. Lobry, Controllability of nonlinear systems on compact manifolds, *SIAM J. Control*, 12, 1–4, 1974.

[11] X. Wang, P. Cui and H. Cui, A note on Poisson stability and controllability, *Nonlinear Dynamics*, 66, 789–793, 2011.

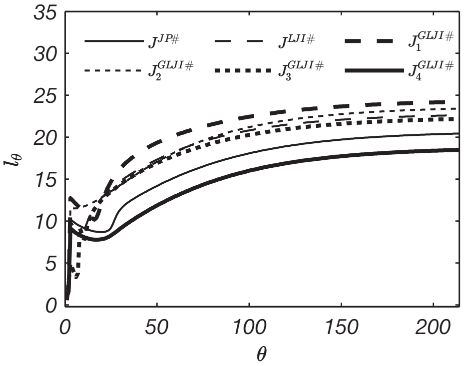

Fig. 5. SRC robot: Euclidean length of the solutions

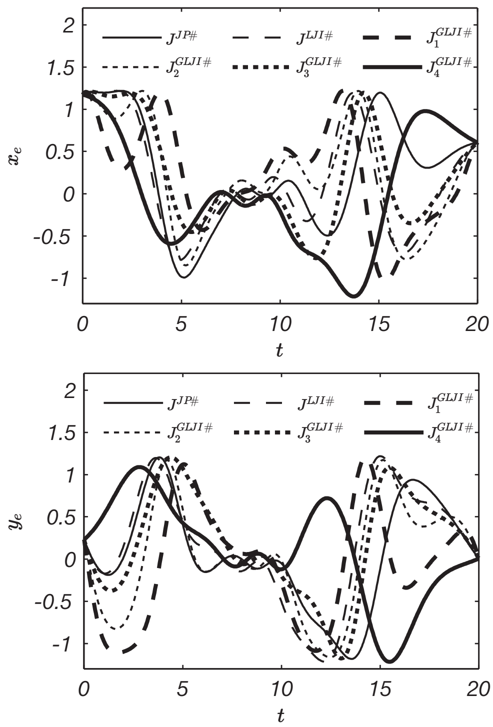

Fig. 6. SRC robot: end-effector trajectories