Robotic Deformable Object Cutting: From Simulation to Experimental Validation*

Philip Long, Wisama Khalil and Philippe Martinet*

***Abstract*—In this paper, investigations into the robotic cutting of soft deformable materials are discussed. A force vision cooperative controller is proposed to deal with the object flexibility and uncertain modeling parameters. The control scheme is firstly tested within a simulator environment and then validated on an experimental cell. The construction of the simulator is described and the interaction with the deformable object discussed. Finally the results of the controller in simulation and experimentation are shown and the differences are discussed.**

I. INTRODUCTION

Robotic control of flexible objects is an important topic and of particular interest in the textile [1], [2], medical [3] and food industry [4], [5]. However in-spite of the increased focus, robotic interaction with deformable materials remains an open issue, chiefly due to the difficulty in predicting the object behavior. This variability can be dealt with in several ways, for example the addition of external sensors to estimate the object state [6], using learning algorithms for repetitive tasks [7] or obtaining a precise object model [8]. Otherwise the object can be modeled as an articulated set of bodies thus allowing classical solutions to be used to control the deformation [9], [10].

The meat processing industry is one of the largest potential markets for deformable object manipulation solutions. A successful adaptation of robotic technology to the meat industry is illustrated in [11] resulting in improved hygiene and precision in the manufacturing environment. In [12] the reduction of the cutting force for a meat cutting robotic cell, by optimizing the cutting feed and cutting angle, is investigated. The ARMS[1] project, *A multi arms Robotic system for Muscle Separation*, aims to contribute to the robotization of the meat industry in particular the separation of beef shoulder muscles. A multi-arm system is proposed in order to deal with key challenges for example the diversity of the target object and its flexibility [13]. The robotic system must complete the same tasks as the human worker i.e. the first arm carries a knife and executes the cutting task while the second is used to open up the object by pulling and tearing at it, finally the third robot carries the camera which is used to obtain the cutting trajectory and update this trajectory as the object deforms.

In this work, a force vision controller is proposed for the cutting task. Force control is required for contact tasks while the vision sensor is ideal for adapting to unknown environments. The combination of force and vision is achieved by either partitioning the task space into force and vision controlled directions [14], [15], or by enforcing impedance relations between them [16], [17]. Generally, force/vision research has been focused on contour following tasks where a force is applied normal to the contour. In cutting applications, this approach is no longer applicable, since in order to separate the object the tool must necessarily pass through the contour.

In this paper, we describe recent investigations into the separation of deformable objects within the framework of the ARMS project. The goal of these investigations is to propose a global control scheme for the meat cutting cell. In order to cope with object flexibility during the separation task, the robotic cell is equipped with an array of exteroceptive sensors, notably force and vision sensors. The construction of a realistic simulator which takes into account the robotic cell, the deformable object and the interaction between them, is outlined. The global controller is tested in the simulator environment before it is transfered to the experimental cell.

The paper is organized in the following way. In Section II, the robotic system is described. In Section III, the simulation of the meat cutting cell including the deformable object is discussed. Furthermore, the global control scheme is outlined. In Section IV the experimental equivalent of the global control scheme is discussed. In Section V, the results are shown. Finally in Section VI, the conclusions are drawn.

II. ROBOTIC SYSTEM

A view of the robotic cell used is given in Fig.1. The simulation system system is composed of three Kuka LWR robots a *cutting* robot, a *pulling* robot and a *vision* robot which will be denoted using the subscripts c, p and v respectively. Tthe main difference between the cells is the absence of a vision robot in the experimental setup. In Section IV, it is demonstrated that a local vision system embedded on the cutting tool can effectively compensate for on-line deformations of the cutting trajectory. The robots are kinematically redundant with 7 revolute joints. The Modified Denavit-Hartenberg (MDH) notation [18] is used to describe the kinematics of the system. From these parameters, for each robot $i = c, p, v$, the following

TABLE I. MDH PARAMETERS OF KUKA ROBOT

j	d	α	θ	$r(m)$
1	0	0	θ_1	0.3105
2	0	$\frac{\pi}{2}$	θ_2	0
3	0	$-\frac{\pi}{2}$	θ_3	0.4
4	0	$-\frac{\pi}{2}$	θ_4	0
5	0	$\frac{\pi}{2}$	θ_5	0.39
6	0	$\frac{\pi}{2}$	θ_6	0
7	0	$-\frac{\pi}{2}$	θ_7	r_7

models are obtained:

$$ {}^0\mathbf{T}_i = \begin{bmatrix} {}^0\mathbf{R}_i & {}^0\mathbf{p}_i \\ 0\ 0\ 0 & 1 \end{bmatrix} \quad (1) $$

*Authors are with Institut de Recherche en Communications et Cybernétique de Nantes, UMR CNRS n° 6597 1 rue de la Noë, LUNAM, École Centrale de Nantes, 44321, Nantes, France. Firstname.Lastname@irccyn.ec-nantes.fr

[1]arms.irccyn.ec-nantes.fr

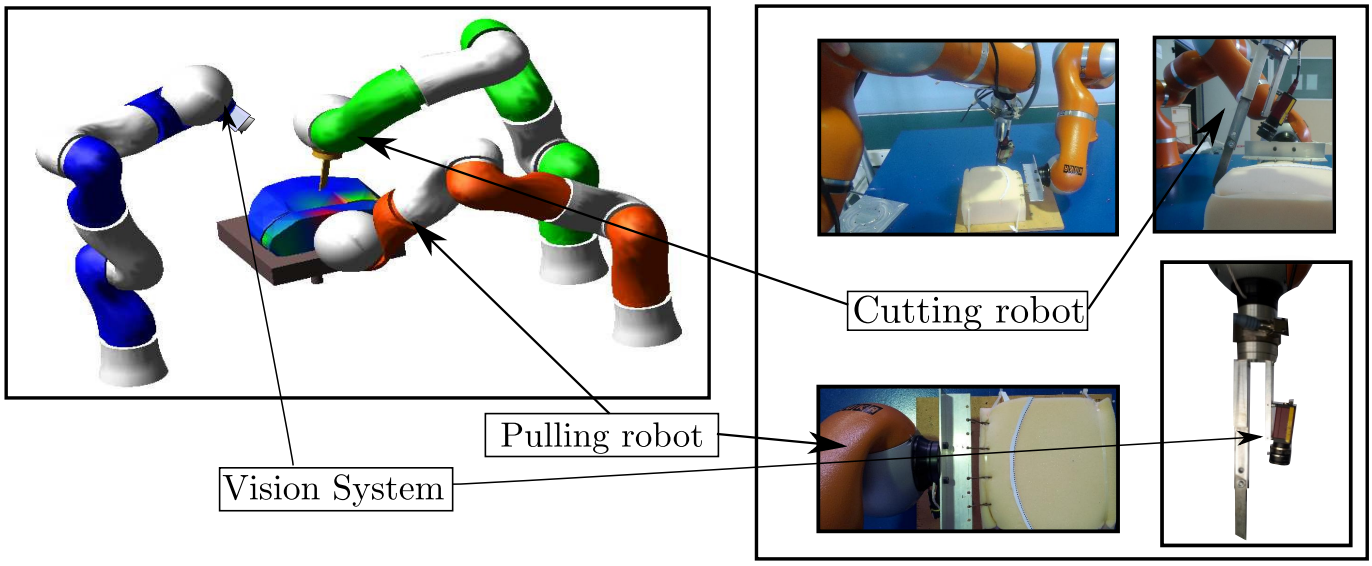

Fig. 1. Comparison of Simulation and Experimental Meat Cutting Cell

$\mathbf{R}_i$ represents the orientation and $\mathbf{p}_i$ the position of the task frame of robot i w.r.t its base frame. Using a minimal representation of orientation $\mathbf{u}\psi_i$, the Cartesian position, kinematic screw, acceleration of robot i in vector form is given as:

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{p}_i \\ \mathbf{u}\psi_i \end{bmatrix} \tag{2}$$

$$\mathbf{V}_i = \mathbf{J}_i \dot{\mathbf{q}}_i \tag{3}$$

$$\dot{\mathbf{V}}_i = \mathbf{J}_i \ddot{\mathbf{q}}_i + \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i \tag{4}$$

The dynamic model of each robot can be written as:

$$\boldsymbol{\tau}_i = \mathbf{A}_i \ddot{\mathbf{q}}_i + \mathbf{H}_i + \mathbf{J}_i^T \mathbf{h}_i \tag{5}$$

$\mathbf{J}_i$ is the kinematic Jacobian matrix and $\mathbf{q}_i$ the vector of joint coordinates while $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are the velocities and accelerations respectively. The inertia matrix and the matrix of centrifugal, Coriolis and gravity torques are denoted as $\mathbf{A}_i$, and $\mathbf{H}_i$. The Cartesian wrench is denoted as $\mathbf{h}_i$ while $\boldsymbol{\tau}_i$ is the joint torque.

III. Simulation of a Meat Cutting Robotic Cell

In this section the simulation of the ARMS robotic cell is described [13]. The simulator is designed so that multiple control schemes that deal with the separation of deformable objects can be tested. Furthermore the simulator allows us to: (i) optimize the position of the robots with respect to the deformable body, (ii) demonstrate the advantages of cell redundancy (iii) demonstrate the advantages of a centralized robotic controller.

A. Simulation of deformable object

The deformable object is at the center of the simulation strategy, it represents the beef shoulder that must be separated. The simulated object must react to pulling forces in a realistic manner. Furthermore the object must be *separable*, i.e. react coherently to the incisions of the robot controlled knife.

To create the deformable object model, the cutting region approach is used [19], [20]. In this case, the knife can interact with the object only within a defined cutting region. As such, two distinct deformable models are generated, a computationally heavy model that allows tool interaction within the cutting region and a computationally efficient model that does not interact with the tool. For the meat cutting application, this approach is convenient since the objective is to separate two beef muscles that are joined by a non-homogeneous region known as the aponeurosis. The aponeurosis represents the cutting region and is located between the muscles. Therefore this region can be modeled specifically for tool interaction.

1) Beef Muscles: In order to create the beef muscles, the following steps, were taken as shown in Fig.2. Firstly a visual scan of a generic beef round is obtained after separation and converted into a 3D-geometry. The two muscles are reconstructed within a CAD program and the exact cutting surface is extracted. Two simplified muscles are created using the exact cutting surface in order to reduce the computational cost during the simulation. The simplified models are discretized volumetrically using a meshing program. A modal analysis is performed for each muscle and the resulting output is a .mnf file.

2) Aponeurosis: The surface of separation of the beef shoulder is distinguished by a set of aponeurosis, that are similar to tendons, acting as links between the main beef muscles. The aponeurosis are modeled as the second deformable object located in an intermediate layer in the beef shoulder. The aponeurosis store elastic energy, then recoil when unloaded. This behavior is approximated as a series of spring damper systems fixed to the muscle at discrete points. Fig.3 gives a visualization of the aponeurosis, as the muscles are been separated.

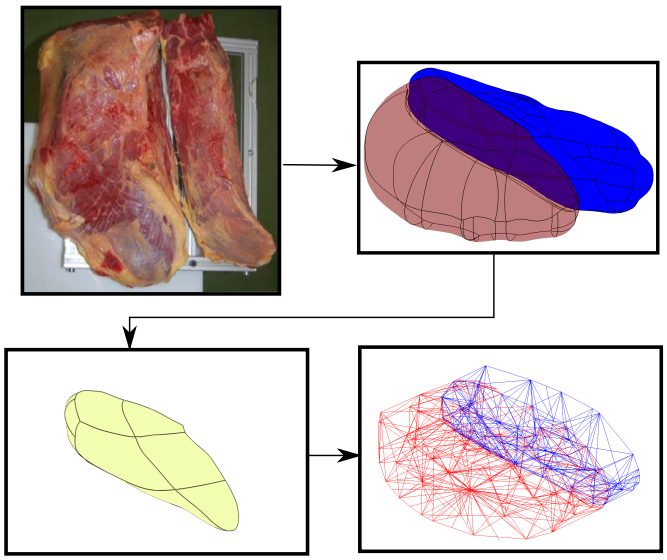

Fig. 2. From 3D scan to Finite element mesh, for clarity the different muscles are shown in blue and red

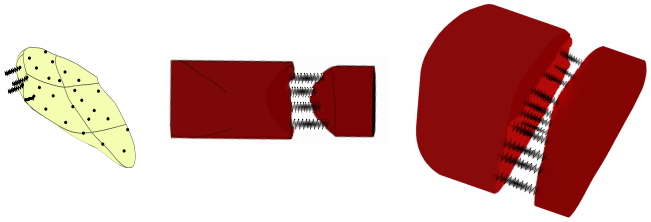

Fig. 3. The Aponeurosis as a set of spring damper system, distributed over the surface on left, after integration between the beef muscles shown in the middle and right images

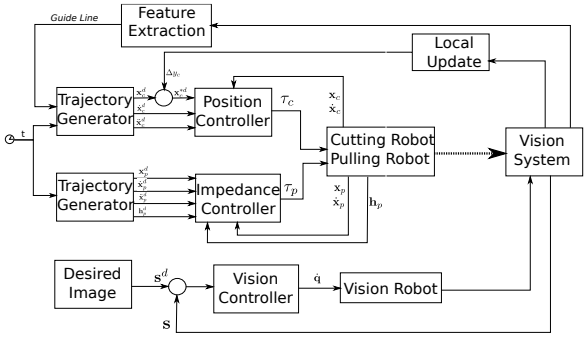

Fig. 4. Global Control Scheme

B. Global Controller

Each arm is controlled independently in their respective tasks while the coupling effects are felt through the interaction with the deformable body. A desired value of a parameter is represented as the same variable with the superscript ^{d}. The prefix Δ denotes the difference between a desired value and the current value of variable, for example $\Delta\mathbf{x}_i = \mathbf{x}_i^d - \mathbf{x}_i$. A global overview of the control scheme is given in Fig. 4.

C. Task Definition

The primary task is the separation of the two meat muscles. To complete this task, the spring-damper links, representing the aponeurosis, must be removed by the passage of the knife. The cutting tool must only interact with the aponeurosis and avoid cutting into the meat muscles at either side. This necessitates a series of cuts, called passages, at increasing depths along the visible guide line. The pulling robot is responsible for creating an opening so that the knife can pass, unobstructed, along the guide line. After each passage the opening will increase allowing the knife to move deeper into the valley until the two objects have been completely separated. The vision robot must alter its pose so that the guide line is kept within the field of view as the meat deforms.

D. Cutting Robot

At the beginning of each passage, the visual primitives are used to reconstruct the guide line. A curve, denoted as $\mathcal{C}$ is then fitted to the guide line. This curve is represented by a polynomial expression. For a given cutting depth z, the desired trajectory is defined by:

$$y = a_2(x)^2 + a_1 x + a_0 \tag{6}$$

The total curvilinear length, D of the polynomial curve is obtained by integrating (7), where a and b are the extremities of the surface.

$$D = \int_b^a \sqrt{1 + \frac{\partial y^2}{\partial x^2}}\, \partial x = |_b^a f(x) \tag{7}$$

$$D = f(a) - f(b) \tag{8}$$

A variable $s(t)$ representing the curvilinear distance along the curve is defined using the temporal constraints (9), (10), (11).

$$s(t=0) = 0 \qquad s(t=t_{final}) = D \tag{9}$$

$$\dot{s}(t=0) = 0 \qquad \dot{s}(t=t_{final}) = 0 \tag{10}$$

$$\ddot{s}(t=0) = 0 \qquad \ddot{s}(t=t_{final}) = 0 \tag{11}$$

At any time t, $s(t)$ is calculated as:

$$s(t) = r(t)D \tag{12}$$

where $r(t)$ is the interpolation function, a 5-DOF polynomial continuous in acceleration given as:

$$r(t) = 10\left(\frac{t}{t_{final}}\right)^3 - 15\left(\frac{t}{t_{final}}\right)^4 + 6\left(\frac{t}{t_{final}}\right)^5 \tag{13}$$

$x(t)$ can be obtained by substituting $s(t)$ into (8), and then solving the following:

$$f(x) = s(t) + f(b) \tag{14}$$

$y(t)$ is calculated from using $x(t)$ the polynomial expression in (6). To complete the cutting task definition, the orientation of the knife must be considered. The orientation is modified such that the cutting side of the knife remains tangent at all times to the polynomial curve.

From the above a desired trajectory is generated in position and orientation, velocity and acceleration, i.e. $\mathbf{x}^d$, $\mathbf{V}^d$ and $\dot{\mathbf{V}}^d$.

To track the desired variables, using Cartesian computed torque the desired Cartesian acceleration,$\mathbf{w}_c$, is defined as:

$$\mathbf{w}_c = \dot{\mathbf{V}}^d + \mathbf{K}_d\left(\Delta\mathbf{V}\right) + \mathbf{K}_p\left(\Delta\mathbf{x}\right) - \dot{\mathbf{J}}_c\dot{\mathbf{q}} \tag{15}$$

where $\mathbf{K}_d$ $\mathbf{K}_p$ are positive gains. $\mathbf{w}_c$ is then transformed to the joint space, and a new desired acceleration exploiting the redundancy of the system is defined

$$\mathbf{w} = \mathbf{J}_c^{+}\left(\mathbf{w}_c + \mathbf{P}_c\ \mathbf{Z}\right) \tag{16}$$

Finally a joint torque realizing this acceleration is obtained

$$\tau_c = \mathbf{A}_c\mathbf{w} + \mathbf{H} \tag{17}$$

However the object deforms during the passage of the knife changing the profile of the cutting surface. This is due both to the force applied by the pulling robot and to the effects of cutting the aponeurosis in the intermediate layer. In order to compensate for this motion, the desired position is updated on-line by using, y_g, the exact position of the guide line extracted from the visual primitive. y^d is updated as:

$$y^{*d}(t) = y^d(t) + \Delta y \tag{18}$$

$$\Delta y = y_g - y_c \tag{19}$$

E. Pulling Robot

To complete the desired task, the pulling robot must be force controlled. The desired behavior is a gradual opening of the cutting valley as the cutting depth increases. For each passage the number of links between the deformable objects is reduced leading to a smaller retaining force. An impedance controller is applied where $\Delta\mathbf{h} = \mathbf{h}_d - \mathbf{h}$ the difference between the desired and current pulling force, where λ is a gain matrix representing the inverse of the desired inertial behavior. Equation (15) is modified to include these terms:

$$\mathbf{w}_p = \dot{\mathbf{V}} + \lambda\left(\mathbf{K}_d\left(\Delta\mathbf{V}\right) + \mathbf{K}_p\left(\Delta\mathbf{x}\right) - \mathbf{K}_f(\Delta\mathbf{h})\right) - \dot{\mathbf{J}}_p\dot{\mathbf{q}} \tag{20}$$

F. Vision Robot

For the vision robot, the task is to maintain the cutting zone in the field of view. To do so the vision robot must be controlled in image space. A desired image, which is defined to optimize the field of view is defined as $\mathbf{s}_d$. The guide line is extracted by the camera and parameterized into a segment feature variable $\mathbf{s}$ as:

$$\mathbf{s} = \begin{bmatrix} x & y & \theta & l \end{bmatrix}^T \tag{21}$$

x, y are the coordinates of the segment's center, θ is the angle and l is the length. The desired velocity is generated using the interaction matrix [17]:

$$\dot{\mathbf{q}}_v^d = -\mathbf{K}_p\left(\mathbf{L}_s\mathbf{J}_v\right)^{+}\left(\mathbf{s}_d - \mathbf{s}_{im}\right) \tag{22}$$

IV. Experimental Validation of Proposed Controller

The global control scheme is also validated experimentally. However in order to do this the following modifications must be made to the vision system. In this case, the vision system is no longer ideal, rather it can only focus within a local zone close to the cutting tool. This means that the curve must be obtained off-line and represented using a sequence of visual markers. The local vision controller is described in the following sections:

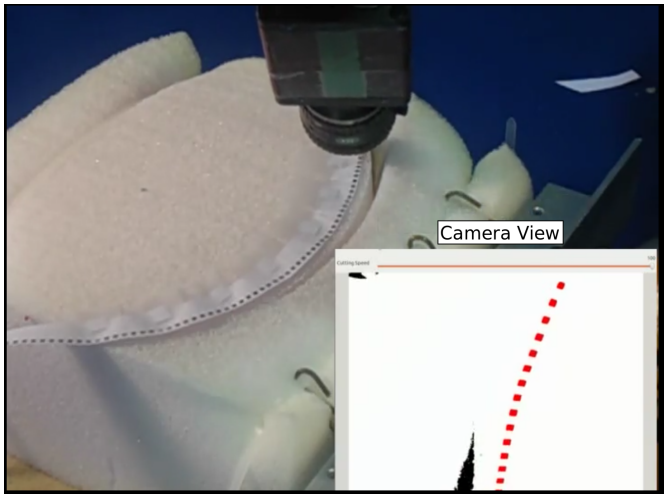

Fig. 5. Robot cutting with camera view

A. Vision Controller

For the experimental validation, the vision system is embedded on the cutting tool as shown in Fig.5. The vision controller updates the trajectory of the knife in response to on-line deformations by creating a deviation, denoted as ${}^{ob}d\mathbf{X}_t^v$ as follows:

1) The vision system extracts the image coordinates of (u_i, v_i), (u_j, v_j) and (u_k, v_k), a series of points ahead of the image projection of tool point
2) The normalized position of a point i is reconstructed using the intrinsic camera parameters, $\mathbf{C}$, which relate the image coordinates to the coordinates in the perspective plane:

$$\begin{bmatrix} \frac{p_{xi}^v}{p_{zi}^v} \\ \frac{p_{yi}^v}{p_{zi}^v} \\ 1 \end{bmatrix} = \mathbf{C}\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \tag{23}$$

3) The depth of a point, p_{zi}^v, is estimated using the material height and the tool position, the depth estimation allows the reconstruction of the 3D position of the point
4) By fitting a straight line to the Cartesian position of points i, j and k the vectors $\mathbf{s}$ and then $\mathbf{n}$ are obtained
5) From step 3 and step 4 the desired matrix given by vision ${}^{ob}\mathbf{T}_i^v$ at a point i can be written

In order to generate an error vector, the curve $\mathcal{C}$ is evaluated at p_{xi} allowing a desired matrix ${}^{ob}\mathbf{T}_i^d$ to be obtained. This in turn is used to calculate the vision generated deviation which acts in one translational direction and three rotational directions:

$$\Delta p_{yi} = p_{yi}^d - p_{yi}^v \tag{24}$$

$$\Delta^{ob}\mathbf{R}_i = {}^{ob}\ \mathbf{R}_i^d\left({}^{ob}\mathbf{R}_i^v\right)^T \tag{25}$$

V. Results & Discussion

Fig.7 shows for the simulated robotic cell: the initial guide before cutting has commenced, the interpolated trajectory for

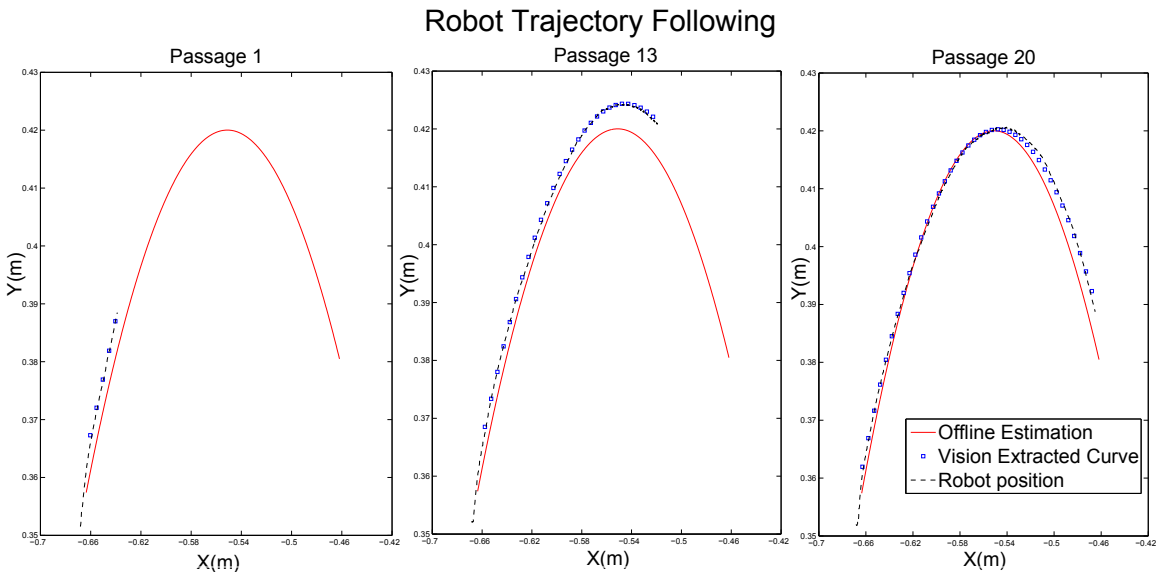

Fig. 6. Robot Trajectory following

this line, the position of the guide line during the cutting trajectory and the position of the robot cutting tool. The results show a large difference between the initial guide line before cutting has begun and the position of the guide line during the trajectory. This is due to the deformation of the object as the links are severed and the continuous application of the pulling force. Fig.7 shows that the visual update described in Section III allows the robot to compensate.

In Fig.6 the off-line estimation of the curve, the visually extracted curve and the robot position are shown for the experimental cell. The graphs show that even though the addition of the pulling force deforms the curve, the vision controller allows the robot to compensate and cut along the new trajectory. Passages 13 and 20 show that the cutting trajectory begins to resemble the initial estimation as the separation nears completion as the object undergoes a relaxation phase.

In both cases, the vision system allows the robot to compensate for on-line deformation. However in the experimental results, the robot does not succeed in cutting the passage length. This is due to the resistive forces encountered during the cut which necessitate a force controller for the cutting robot.

VI. Conclusion

In this paper, robotic controlled separation of deformable objects has been discussed. A force vision cooperative control scheme has been proposed that allows the system to simultaneously hold the object and locate and cut along a deformable trajectory. The control scheme is validated first in simulation then experimentally.

Future work will focus on centralized redundancy resolution of the robotic cell. The aim is to use the supplementary DOF of each robot to accomplish secondary objectives. Furthermore a force controller will be derived for the cutting robot in order to avoid a build-up of large resistive forces.

Acknowledgments

This paper describes work carried out in the framework of the ARMS project. ARMS : A multi arms robotic system for muscle separation, is partly funded by the ANR (Agence Nationale de la Recherche), reference ANR-10-SEGI-000.

References

[1] M. Kudo, Y. Nasu, K. Mitobe, and B. Borovac, "Multi-arm robot control system for manipulation of flexible materials in sewing operation," *Mechatronics*, vol. 10, no. 3, pp. 371–402, 2000.

[2] M. Saadat and P. Nan, "Industrial applications of automatic manipulation of flexible materials," *Industrial Robot: An International Journal*, vol. 29, no. 5, pp. 434–442, 2002.

[3] K. Cleary and C. Nguyen, "State of the art in surgical robotics: clinical applications and technology challenges," *Computer Aided Surgery*, vol. 6, no. 6, pp. 312–328, 2001.

[4] R. J. M. Masey, J. O. Gray, T. J. Dodd, and D. G. Caldwell, "Guidelines for the design of low-cost robots for the food industry," *Industrial Robot: An International Journal*, vol. 37, no. 6, pp. 509–517, 2010.

[5] S. Tokumoto and S. Hirai, "Deformation control of rheological food dough using a forming process model," in *Proceedings of 2002 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1457–1464, IEEE, 2002.

[6] D. Navarro-Alarcón, Y.-H. Liu, J. G. Romero, and P. Li, "Model-free visually servoed deformation control of elastic objects by robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 29, pp. 1457–1468, dec 2013.

[7] B. Balaguer and S. Carpin, "Combining imitation and reinforcement learning to fold deformable planar objects," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1405–1412, IEEE, 2011.

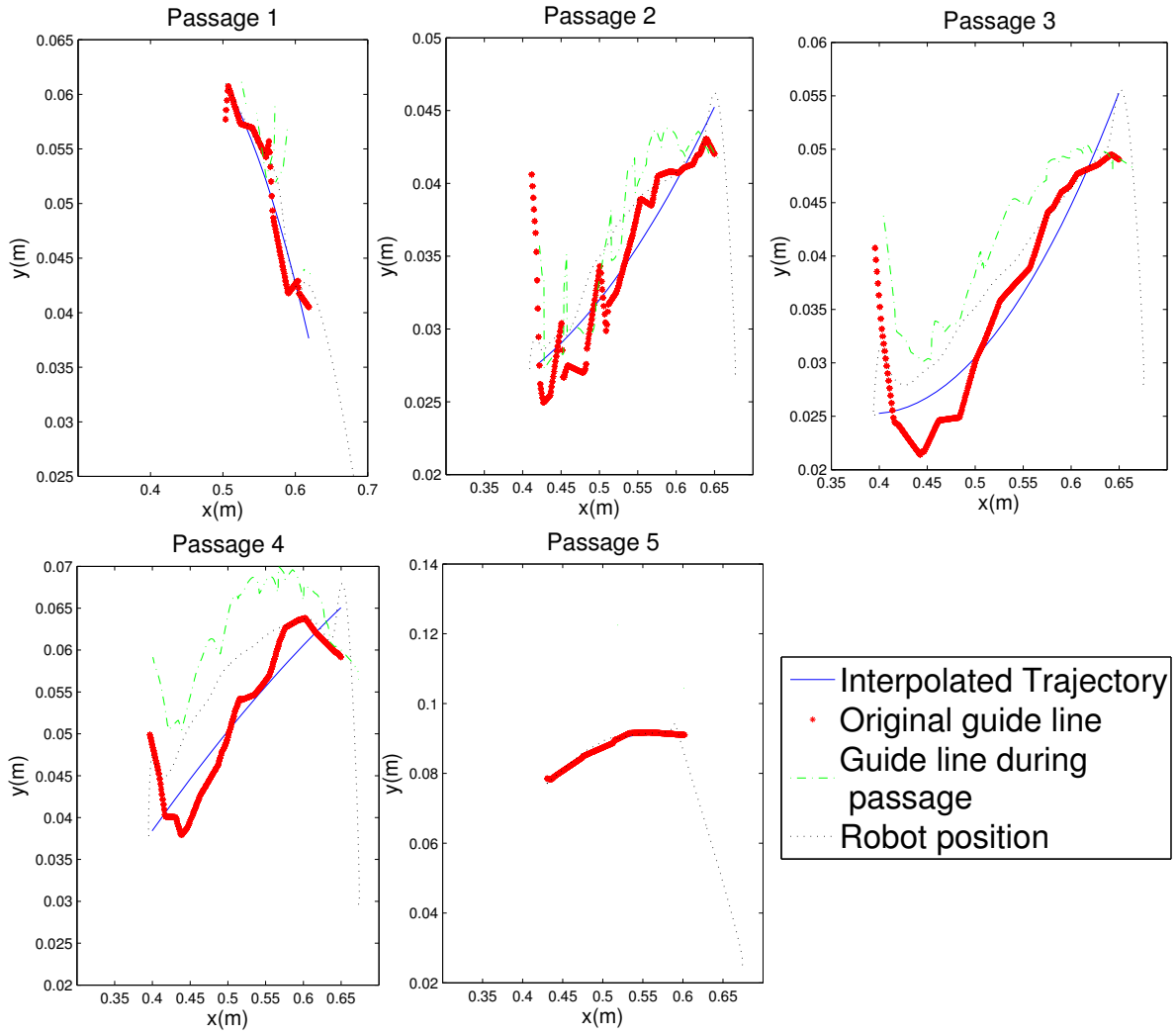

Fig. 7. Robot Trajectory modified by local updates

[8] Y. Luo and B. J. Nelson, "Fusing force and vision feedback for manipulating deformable objects," *Journal of Robotic Systems*, vol. 18, no. 3, pp. 103–117, 2001.

[9] W. F. Dellinger and J. Anderson, "Interactive force dynamics of two robotic manipulators grasping a non-rigid object," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pp. 2205–2210 vol.3, 1992.

[10] D. Weer and S. Rock, "Experiments in object impedance control for flexible objects," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp. 1222–1227, IEEE, 1994.

[11] L. Hinrichsen, "Manufacturing technology in the danish pig slaughter industry," *Meat science*, vol. 84, no. 2, pp. 271–275, 2010.

[12] G. Guire, L. Sabourin, G. Gogu, and E. Lemoine, "Robotic cell for beef carcass primal cutting and pork ham boning in meat industry," *Industrial Robot: An International Journal*, vol. 37, no. 6, pp. 532–541, 2010.

[13] P. Long, W. Khalil, and P. Martinet, "Modeling and control of a meat-cutting robotic cell," in *2013 15th International Conference on Advanced Robotics (ICAR)*, pp. 61–66, IEEE, 2013.

[14] K. Hosoda, K. Igarashi, and M. Asada, "Adaptive hybrid control for visual and force servoing in an unknown environment," *Robotics & Automation Magazine, IEEE*, vol. 5, no. 4, pp. 39–43, 1998.

[15] D. Xiao, B. Ghosh, N. Xi, and T. Tarn, "Sensor-based hybrid position/force control of a robot manipulator in an uncalibrated environment," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 635–645, 2000.

[16] V. Lippiello, B. Siciliano, and L. Villani, "Robot interaction control using force and vision," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 1470–1475, IEEE, 2006.

[17] Y. Mezouar, M. Prats, and P. Martinet, "External hybrid vision/force control," in *Intl. Conference on Advanced Robotics (ICAR07)*, 2007.

[18] W. Khalil and J. Kleinfinger, "A new geometric notation for open and closed-loop robots," in *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1174–1179, IEEE, 1986.

[19] H. Delingette, S. Cotin, and N. Ayache, "A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation," in *Computer Animation, 1999. Proceedings*, pp. 70–81, 1999.

[20] L. Vigneron, J. Verly, and S. Warfield, "On extended finite element method (xfem) for modelling of organ deformations associated with surgical cuts," *Medical Simulation*, pp. 134–143, 2004.