On the Multifrequency Calculation of Robot Trajectories

J. A. Tenreiro Machado
Faculty of Engineering of the University of Porto
Dept. of Electrical and Computer Engineering
R. Bragas, 4099 Porto Codex, Portugal

Carla S. M. Cordeiro
Modern University,
Dept. of Automation and Control Engineering
Rua Augusto Rosa, 24, 4000 Porto, Portugal

Imre J.Rudas
Bánki Donát Polytechnic
Dept. of Information Technology
H-1081 Budapest, Népszínház utca 8
Hungary

Abstract — The computational burden of robot trajectory planning algorithms may be significant, namely when the inverse kinematics is required. Nevertheless, the variables involved (*e.g.* positions, velocities and accelerations) have distinct bandwidths. Therefore, they have different effects upon the final result and we can adopt a multifrequency strategy for the calculations, namely using different frequencies for the robot joint positions, velocities and accelerations. In this line of thought, this paper compares a multirate trajectory planning scheme in the cartesian space in contrast with the classical monofrequency calculation strategy.

I. INTRODUTION

In the last decades robotics has been a major area of research and development in industrial automation [1-5]. The kinematics, dynamics and control [6-8] of mechanical manipulators are presently standard issues when thinking on practical implementations. Nevertheless, these algorithms have aspects and details still unexplored that deserve further attention.

Bearing these facts in mind, this paper studies the multirate computation of the robot kinematics, namely its influence on the trajectory evolution and the associated processing time. This strategy is motivated by the different bandwidths of the signals involved (*i.e.*, positions, velocities and accelerations) that lead to distinct effects upon the resulting trajectory. However, the non-linear nature of the kinematic equations makes difficult to predict the numerical errors due to a finite sampling rate. Consequently, the definition of a optimal multifrequency scheme, establishing a compromise with a given computational load, must be analyzed. In fact, some authors have proposed different calculation frequencies [9] for the signals without considering, neither distinct multirate schemes, nor the effects upon the final results. Therefore, a deeper investigation must evaluates all the sides of the problem, before establishment of a given computer implementation.

In this perspective the paper is organized as follows. Section 2 starts by defining the statistical measures for the analysis of robot trajectory errors arising in a finite sampling evaluation. Section 3 presents the other side of the problem, that is, the computational burden associated with the kinematic algorithm. Section 4 develops optimization methods for a compromise between the trajectory errors and the corresponding computational load. Finally, section 5 outlines the main conclusions.

II. TRAJECTORY ERRORS

The goal of this study consists in determining a compromise between the trajectory errors for the positions, velocities and accelerations and the corresponding computational burden, when adopting different sampling frequencies.

The work adopts a two link planar robot arm (RR) (Fig. 1) and straight-line trajectories, assuming rectangular on-off acceleration profiles. The trajectories are defined in the Cartesian space, which implies the calculation of the inverse kinematics to obtain the corresponding joint values.

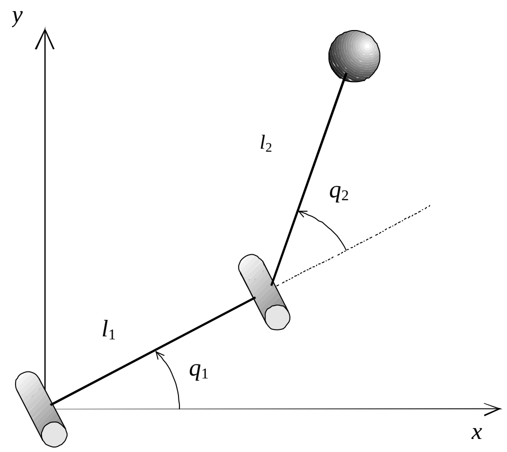
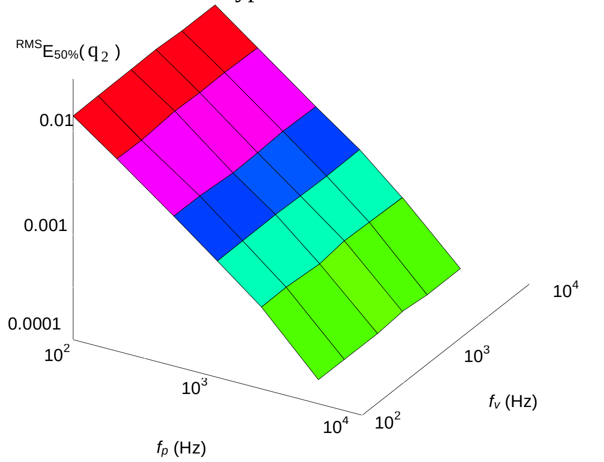


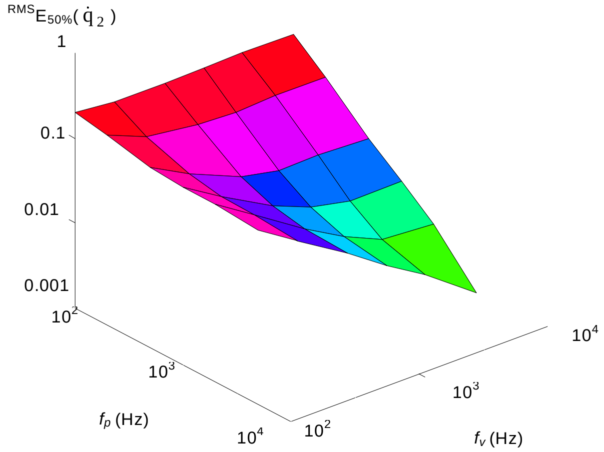
Fig. 1: The RR robot manipulator.

Due to the infinite number of possible robot trajectories, it is used a statistical perspective to determine the root-mean-square (RMS) errors for different calculation frequencies. The statistical results of a random sample of distinct trajectories are presented in percentiles, in detriment of others statistical measures, due to their robustness for experiments passing near singular points. In this perspective, it is adopted the notation ${}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(q_i)$, ${}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\dot{q}_i)$ and ${}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\ddot{q}_i)$ for the P-percentiles of the RMS-errors for the *i*th joint position, velocity and acceleration, respectively. Furthermore, it is also investigated the influence of different robot link lengths in order to determine their influence on the results.

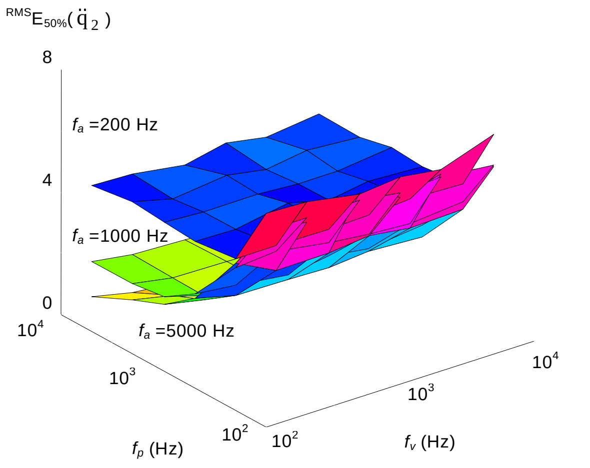
Fig. 2 shows a typical example of the results, namely the 50%-percentiles of the RMS trajectory errors for the joint 2 position, velocity and acceleration *versus* the calculation frequencies. For other percentiles of the RMS errors we have charts of the same type.



a)



b)



c)

Fig 2: Percentile of 50% of the RMS error for joint 2 (l_1=1.8 m, l_2=0.2 m)
a) Position
b) Velocity
c) Acceleration

As expected, the results demonstrate that the RMS error for the positions, velocities and accelerations depend only on $\{f_p\}$, $\{f_v, f_p\}$ and $\{f_a, f_v, f_p\}$, respectively. In the present case, we can obtain numerically heuristic formulae for the errors such as:

$$ {}^{\mathrm{RMS}}\mathrm{E}_{50\%}(q_i) = \frac{K_{pi}}{f_p} \quad (1.a)$$

$$ {}^{\mathrm{RMS}}\mathrm{E}_{50\%}(\dot{q}_i) = \frac{K'_{pi}}{f_p} + \frac{K'_{vi}}{f_v} \quad (1.b)$$

$$ {}^{\mathrm{RMS}}\mathrm{E}_{50\%}(\ddot{q}_i) = \frac{K''_{pi}}{f_p} + \frac{K''_{vi}}{f_v} + \frac{K''_{ai}}{\sqrt{f_a}} \quad (1.c)$$

where $K_{pi}, \cdots, K''_{ai}$ are system dependent constants for the ith robot joint. The expressions for other percentiles are similar except for the percentile of 0% due to numerical errors resulting from very small values.

The parameters $K_{pi}, \cdots, K''_{ai}$ depend on the arm link lengths l_1 and l_2. For example, for the 50% percentile we have the approximate expressions:

$$K_{p1} = 1.10 - 0.63\frac{l_1 - l_2}{l_1 + l_2} \quad (2.a)$$

$$K_{p2} = 1.722 - 0.247(l_1 + l_2) \quad (2.b)$$

$$K'_{p1} = \frac{7.28}{l_1 + l_2} - 4\frac{|l_1 - l_2| + (l_1 - l_2)}{(l_1 + l_2)^2} \quad (2.c)$$

$$K'_{v1} = \frac{13}{l_1 + l_2} - 8\frac{l_1 - l_2}{(l_1 + l_2)^2} \quad (2.d)$$

$$K'_{p2} = \frac{18}{l_1 + l_2} + 18\frac{l_1 - l_2}{(l_1 + l_2)^2} \quad (2.e)$$

$$K'_{v2} = \frac{11}{l_1 + l_2} + 311\frac{(l_1 - l_2)^4}{(l_1 + l_2)^5} \quad (2.f)$$

The expressions for other cases are under study.

III. COMPUTATIONAL BURDEN *VERSUS* TRAJECTORY ERRORS

In a multifrequency scheme a compromise between the trajectory errors and the computational burden [10] determine an 'optimal' distribution. For example, in a first experience, Figs. 3 to 5 compare the time evolution of the joint positions, velocities and accelerations for a monofrequency calculation (with $f_a = f_v = f_p = 10^3$ Hz) *versus* a multifrequency scheme (with $f_a = 10^3$ Hz, $f_v = 200$ Hz, $f_p = 10^2$ Hz).

Due to the low sampling frequencies the trajectory errors are significant and may not be acceptable. This means that there is a need for speeding-up the calculation rate.

In a second experiment, Figs. 6 and 7 compare the position and velocity trajectories for the monofrequency processing *versus* a new multifrequency distribution with

f_a = 500 Hz, f_v = 250 Hz, f_p = 200 Hz. Obviously, in this case we have smaller trajectory errors but an higher computational load.

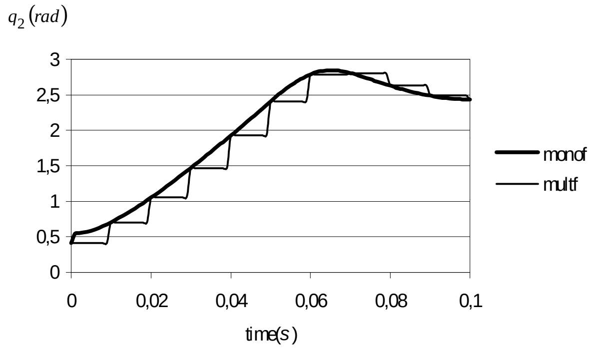


Fig. 3: Time evolution of the joint 2 position ($l_1 = l_2$ = 1.0 m)
Monofrequency: $f_a = f_v = f_p = 10^3$ Hz
Multifrequency: f_a = 500 Hz, f_v = 200 Hz, $f_p = 10^2$ Hz

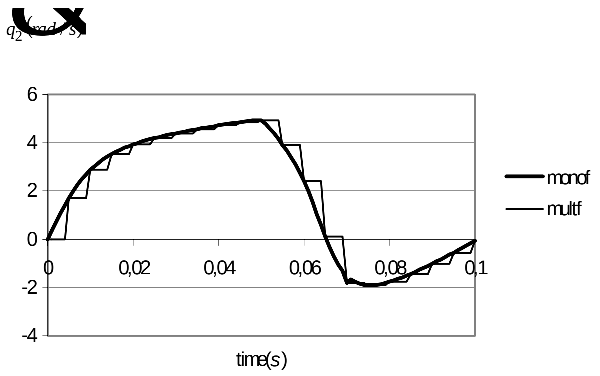


Fig. 4: Time evolution of the joint 2 velocity ($l_1 = l_2$ = 1.0 m)
Monofrequency: $f_a = f_v = f_p = 10^3$ Hz
Multifrequency: f_a = 500 Hz, f_v = 200 Hz, $f_p = 10^2$ Hz

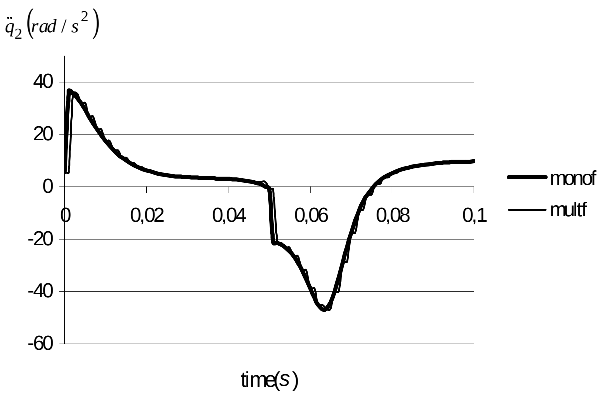


Fig. 5: Time evolution of the joint 2 acceleration ($l_1 = l_2$ = 1.0 m)
Monofrequency: $f_a = f_v = f_p = 10^3$ Hz
Multifrequency: f_a = 500 Hz, f_v = 200 Hz, $f_p = 10^2$ Hz

For the three cases the total computational time T for a 80486DX2@100MHZ is:

- Monofrequency, $f_a = f_v = f_p = 10^3$ Hz
  $T = 16.3\times10^{-3}$ sec
- Multifrequency, f_a=500 Hz, f_v=200 Hz, f_p=100 Hz
  $T = 4.2\times10^{-3}$ sec
- Multifrequency, f_a=500 Hz, f_v=250 Hz, f_p=200 Hz
  $T = 5.0\times10^{-3}$ sec

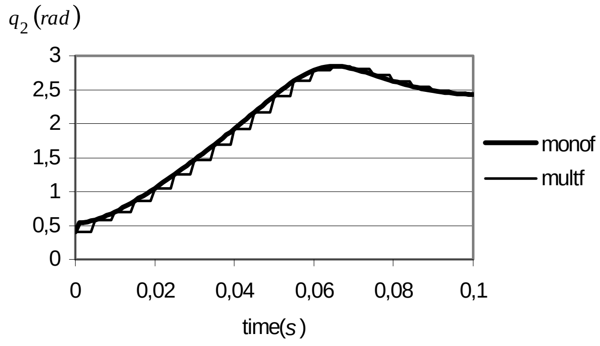


Fig. 6: Time evolution of the joint 2 position ($l_1 = l_2$ = 1.0 m)
Monofrequency: $f_a = f_v = f_p = 10^3$ Hz
Multifrequency: f_a = 500 Hz, f_v = 250 Hz, f_p = 200 Hz

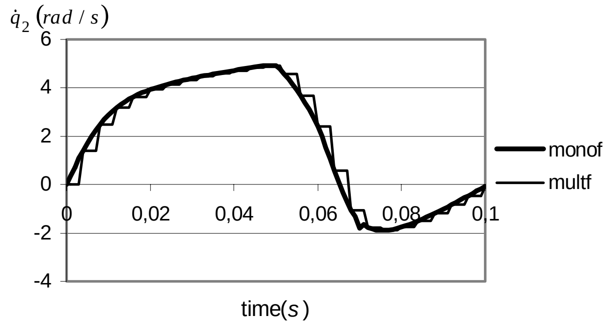


Fig. 7: Time evolution of the joint 2 velocity ($l_1 = l_2$ = 1.0 m)
Monofrequency: $f_a = f_v = f_p = 10^3$ Hz
Multifrequency: f_a = 500 Hz, f_v = 250 Hz, f_p = 200 Hz

The importance of multirate schemes is related with the computational time saving. In this perspective, we need to identify the mathematical operations involved in the inverse kinematics and to cluster them in different groups according to its processing time. According with [10] the groups are {multiplication/division, sum/subtraction, sine/cosine, arctang, square root}. Each one of these groups has a different processing time resulting, for the multifrequency scheme, an average total calculation time T_{av} for each trajectory point (*i.e.*, for a point in the position, velocity and acceleration trajectories) given by an expression:

$$T_{av} = \frac{1}{\max(f_a, f_v, f_p)} \left\{ f_a \sum_i K_a^i\, T_i + f_v \sum_j K_v^j\, T_j + f_p \sum_k K_p^k\, T_k \right\} \tag{3}$$

where T_i are processing times for different floating point operations and K_a^i, K_v^i, K_p^i are the corresponding number of operations involved. Consequently, the sampling period

T_s adopted for the highest calculation frequency in the multifrequency scheme must be:

$$T_s \ge T_{av} \tag{4}$$

Using a 80486DX2 computer, with a frequency clock of 100 MHz, we get:

$$T_a = \sum_i K_a^i \, T_i \approx T_v = \sum_j K_v^j \, T_j = 4.8\times10^{-6} \; sec$$
$$T_p = \sum_k K_p^k \, T_k = 3.77\times10^{-5} \; sec \tag{5}$$

Due to the reduced computational load of the RR-robot kinematics, for simulation purposes it is considered that for calculations we have merely 10% of the computer processing capability presented in (5).

IV. OPTIMIZATION

Our main concern is to determine a set of values for f_a, f_v and f_p that guarantee on one side small errors for the positions, velocities and accelerations and, on the other side a low calculation time. Those goals are obviously contradictory. Therefore we have a problem with multiple goals and there is no unique optimal solution.

For a P-percentile our multi-objective problem can be formulated as follows (i = 1, 2):

$$\begin{aligned} &\underset{i}{Min} \left\{ {}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(q_i) \right\} \\ &\underset{i}{Min} \left\{ {}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\dot{q}_i) \right\} \\ &\underset{i}{Min} \left\{ {}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\ddot{q}_i) \right\} \\ &Min \left\{ T_s \right\} \end{aligned} \tag{6}$$

subjected to the constraints:

$$f_a, f_v, f_p \ge 0 \tag{7a}$$
$$T_s \ge T_{av} \tag{7b}$$

The methodology used to solve this problem was based on the minimization of an aggregated cost index, J_k, ($k \in \aleph$) defined as

$$J_k = \sum_i w_{pi} \left[{}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(q_i) \right]^k + \sum_i w_{vi} \left[{}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\dot{q}_i) \right]^k + \sum_i w_{ai} \left[{}^{\mathrm{RMS}}\mathrm{E}_{\mathrm{P}}(\ddot{q}_i) \right]^k + w_t \, T_s \tag{8}$$

where w_{pi}, w_{vi}, w_{ai} and w_t are weighting factors, that is, priorities associated to each objective. Therefore, the new problem can be formulated as follows:

$$Min\{J_k\}$$
$$\text{Subject to: } f_a, f_v, f_p \ge 0 \tag{9}$$

$$T_s \ge T_{av}$$

To solve this problem it was adopted a numerical method based on successive approximations. In this line of thought, the admissible frequency range is divided in a discrete grid, with increasing accuracy and closer upper/lower limits for successive iterations, until converging to an optimal solution in the sense of (9). During any iteration of the algorithm, if the candidate solution is located at the grid border then it is not guaranteed the algorithm convergence and, therefore, we must enlarge the frequency limits. The flowchart of the algorithm is represented in Fig. 8.

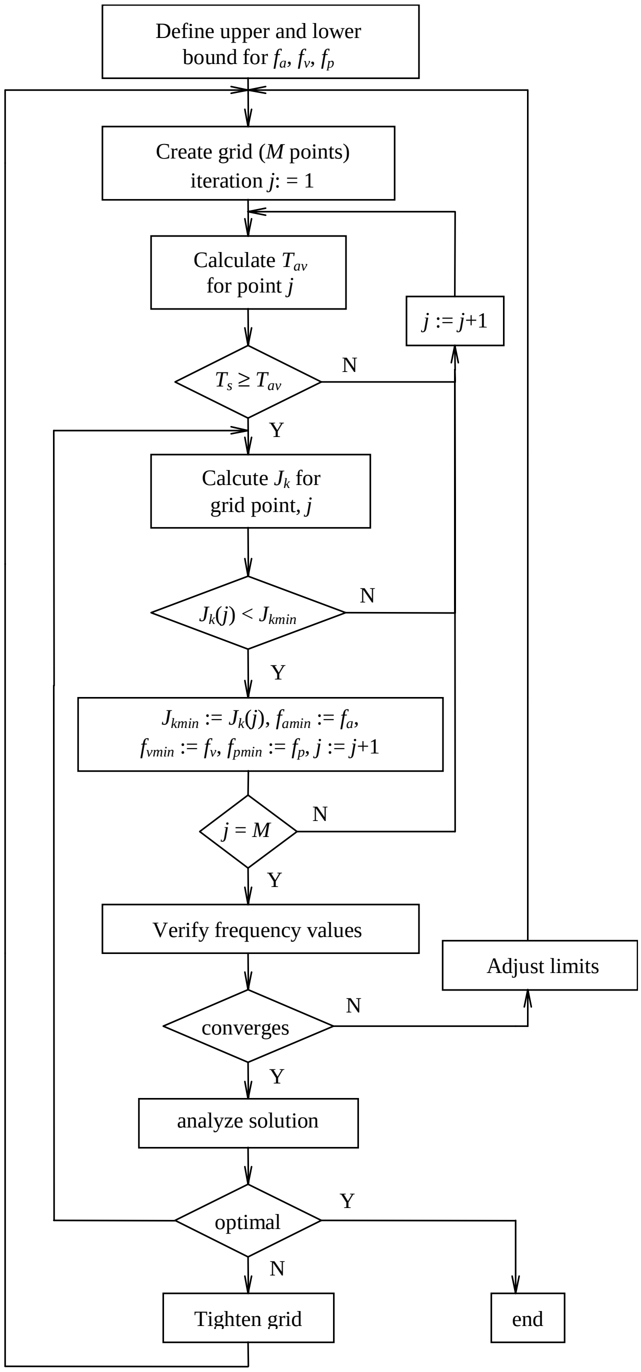


Fig. 8: Flow chart of the optimization algorithm

Several situations were investigated in order to determine the optimal calculation frequencies $\{f_a, f_v, f_p\}$ for different weighting factors in (8). The results indicate that in all cases the multifrequency strategy should obey the relationship $f_a \geq f_v \geq f_p$. This result is easily explained in a signal processing perspective, since the accelerations may suffer sudden variations while the positions are the smoothest [11,12].

The minimization of T_s and the minimization of the trajectory errors have opposite effects. Therefore, we analyze (Figs. 9 to 12) the evolution of the cost index J_k (for k=1 and k=2) and the corresponding optimal solution in terms of $\{f_a, f_v, f_p\}$ for weighting factors $w_{pi}=w_{vi}=w_{ai}=1$ and $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ and $w_t > 0$.

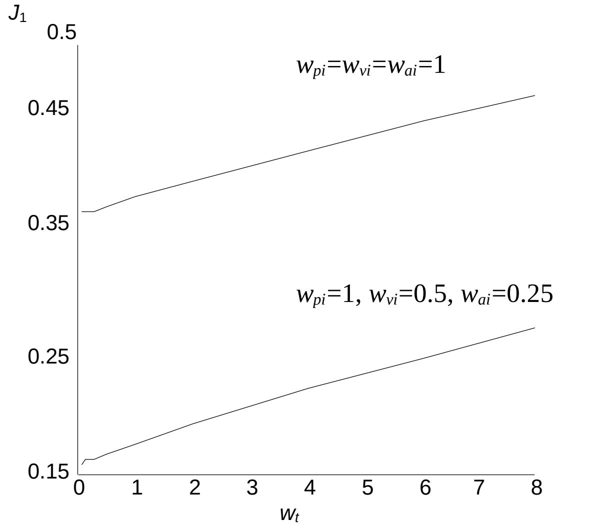


Fig 9: Evolution of the cost index J_1 for $w_{pi}=w_{vi}=w_{ai}=1$ and $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ with $w_t > 0$.

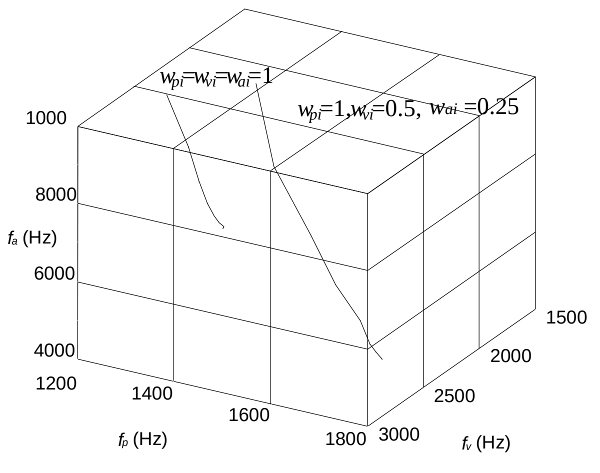


Fig 10: Locus of the optimal solution $\{f_a, f_v, f_p\}$ for J_1 and for $w_{pi}=w_{vi}=w_{ai}=1$ and $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ with $w_t > 0$.

In all cases we get $f_a \geq f_v \geq f_p$ which is in accordance with the intuition of assigning higher sampling frequencies the higher the bandwidth. However, we do not have fixed ratios f_v / f_a and f_p / f_a as proposed by some authors and, in fact, the results are:

- For $w_{pi}=w_{vi}=w_{ai}=1$ and $k = 1$
  $f_a / f_v = 2.8$, $f_a / f_p = 5.1$ ($w_t = 1/16$)
  $f_a / f_v = 4.2$, $f_a / f_p = 7.7$ ($w_t = 8$)
- For $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ and $k = 1$
  $f_a / f_v = 1.8$, $f_a / f_p = 2.7$ ($w_t = 1/16$)
  $f_a / f_v = 4.8$, $f_a / f_p = 7.3$ ($w_t = 8$)
- For $w_{pi}=w_{vi}=w_{ai}=1$ and $k = 2$
  $f_a / f_v = 4.6$, $f_a / f_p = 7.4$ ($w_t = 1/16$)
  $f_a / f_v = 5.0$, $f_a / f_p = 8.0$ ($w_t = 16$)
- For $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ and $k = 2$
  $f_a / f_v = 3.5$, $f_a / f_p = 4.8$ ($w_t = 1/16$)
  $f_a / f_v = 4.7$, $f_a / f_p = 6.6$ ($w_t = 16$)

We conclude that both J_1 and J_2 have a linear dependence with w_t, with similar slopes for the two sets of weighting factors $\{w_{pi}, w_{vi}, w_{ai}\}$.

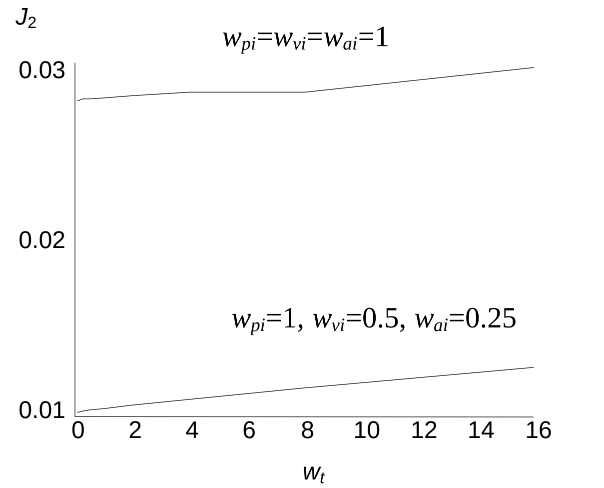


Fig 11: Evolution of the cost index J_2 for $w_{pi}=w_{vi}=w_{ai}=1$ and $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ with $w_t > 0$.

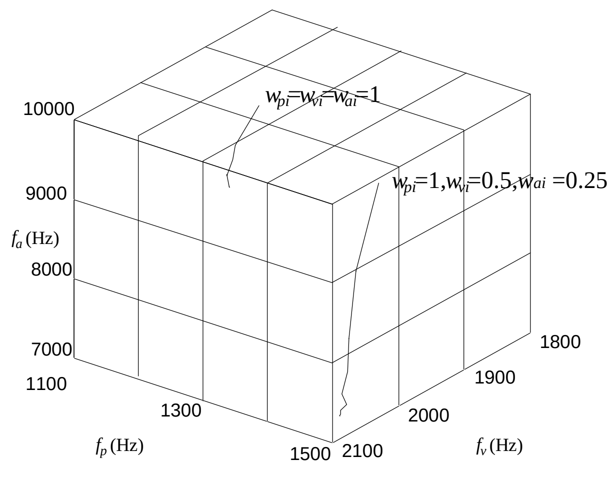


Fig 12: Locus of the optimal solution $\{f_a, f_v, f_p\}$ for J_2 and for $w_{pi}=w_{vi}=w_{ai}=1$ and $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$ with $w_t > 0$.

In what concerns the locus of optimal frequencies $\{f_a, f_v, f_p\}$, in the sense of J_1 and J_2, we have also nearly straight lines. Nevertheless, for $w_{pi}=w_{vi}=w_{ai}=1$ the locus reveals a smaller range of variation than the one corresponding to $w_{pi}=1$, $w_{vi}=0.5$, $w_{ai}=0.25$. This means that in this case we have a higher sensitivity. Moreover, comparing the results for the optimization with $k = 1$ and $k = 2$, we observe that the second case is more conservative.

V. CONCLUSIONS

The effect of the sampling frequency upon the performance of a computer control system is very important. For a robot controller having several algorithms and models it is natural to expect distinct speeds for each case and, therefore, it is reasonable to allocate them different sampling and computing rates. In this perspective, for a given algorithm, the multirate scheme assigns the computing power to each variable in accordance to its needs leading to a more rational management of the system resources. In this paper this strategy was investigated in the calculation of the inverse kinematics for the RR robot trajectories. A compromise between accuracy and computational burden lead to the development of a simple optimisation algorithm. The results are consistent with intuition of assigning higher calculation frequencies the higher the signal bandwidth while giving a formal basis for the problem definition.

VI. REFERENCES

[1] Richard P. Paul, *Robot Manipulators: Mathematics, Programming and Control*, The MIT Press, 1981.

[2] K. S Fu.; R. C Gonzalez; C. S. G. Lee, “Robotics: Control, Sensing, Vision and Intelligence”, McGraw-Hill, 1987.

[3] Saïd M. Megahead, “Principles of Robot Modelling and Simulation”, John Wiley & Sons, 1993.

[4] J. Robert Schilling, “Fundamentals of Robotics: Analysis and Control”, Prentice-Hall, 1990.

[5] Mark Spong, W.; M. Vidyasagar, “Robot Dynamics and Control”, John Wiley & Sons, 1989.

[6] J.K. Tar, I.J. Rudas, J.F. Bitó and M. Dinev, “Adaptive Tuning in an Extended Parameter Space in a Robot Control Based on the Hamiltonian Mechanics”. In Proceedings of the *IEEE Internacional Conference on Intelligent Engineering Systems*, Budapest, Hungary, September 1997, pp. 183-188.

[7] J.K. Tar, I.J. Rudas, J.F. Bitó and O.M. Kaynak, “Constrained Canonical Transformations Used for Representing System Parameters in an Adaptive Control of Manipulators”, In Proceedings of the *8th Internacional Conference on Advanced Robotics*, Monterey, California, U.S.A, July 1997, pp. 873-878.

[8] László Horváth, Imre. J. Rudas, “Procedures for Generating and Evaluation of Generic Manufacturing Process Model Entities”. In Proceedings of the *1997 Internacional Conference on System, Man and Cybernetics*, Orlando, USA, October 1997, pp. 565-570.

[9] M. Kircanski, M. Vukobratovic, N. Kircanski and T. Timcenko, “A New Program Package for the Generation of Efficient Manipulator Kinematic and Dynamic Equations in Symbolic Form”. *Robotica*, 1988, vol. 6, pp. 311-318.

[10] J.A. Tenreiro Machado and Alexandra M. Galhano, “Benchmarking Computer Systems for Robot Control”. *IEEE Trans. on Education*, Aug. 1995, vol. 38, no. 3, pp. 205-210.

[11] J.A. Tenreiro Machado and J.L. Martins Carvalho, “Engineering Design of a Multirate Nonlinear Controller for Robot Manipulators”, *Journal of Robotic Systems*, vol. 6, n. 1, John Wiley & Sons, 1989.

[12] J.A. Tenreiro Machado e Abílio Azenha, “Variable Structure Position/Force Hybrid Control of Manipulators”, In Proceedings of *AARTC’97-4th IFAC Workshop on Algorithms and Architectures for Real-Time Control*, April 1987, Vilamoura, Portugal, pp. 347-352.