Preemptive Scheduling of Multi-rod Hydraulic Infinite Linear Actuator

Robert Braun , Magnus Sethsson , Magnus Landberg*, and Petter Krus

Department of Management and Engineering, Linköping University, Sweden
E-mail: robert.braun@liu.se, magnus.landberg@saabgroup.com, magse13@gmail.com
*Saab Aeronautics, Linköping, Sweden

Abstract

The Hydraulic Infinite Linear Actuator (HILA) is an actuator with a through-rod design, where the piston is able to grip or release the rod during operation. If multiple pistons, connected in series, take turn in gripping, moving, and releasing the rod, a continuous linear motion without the stroke limitations of conventional hydraulic actuators is achieved. This allows the actuator to extend or retract indefinitely while maintaining force continuity. The HILA system can be extended so that each piston is able to grip or release multiple rods, called HILA Multi-Rod (HILA MR). In this way, a few hydraulic pistons are able to control the motion of several rods. In order to determine which rod should be moved by which piston and when, a scheduling algorithm is required. This paper investigates two scheduling algorithms: time-domain multiplexing and round-robin scheduling. The round-robin algorithm is extended with an early-return feature and greedy job selection. All algorithms are evaluated on a simulation model of a HILA with three rods and one or two pistons running a reference cycle. The simulation results show that greedy job selection outperforms the other algorithms, both with one and two servos. As a final test, the scheduler was extended with the ability to give priority to one of the rods. The results show that the rod with priority performs very well and that the priority feature is fairly easy to implement.

Keywords: Actuator, piston, control, scheduling

1 Introduction

The Hydraulic Infinite Linear Actuator (HILA) is a specialized mechanism that enables continuous linear motion without the stroke limitations of conventional hydraulic actuators [1]. This is achieved through a system of interlocking hydraulic pistons that alternates between gripping and releasing the piston rod in a synchronized fashion. This allows the actuator to extend or retract indefinitely while maintaining force continuity. HILA combines the power density of hydraulic actuators with the benefits of a long stroke, enabling for compact and efficient solutions. Experiments have been conducted on applications in various fields, including agricultural and forest machinery [2], elevators in mid-rise buildings [3], and not least aerospace applications [4] where weight and volume reduction is of great importance. In this paper, four different scheduling algorithms for controlling three rods with one or two hydraulic pistons are evaluated against a simulation model.

In its simplest form, the HILA system, with a single piston rod, utilizes a hydraulic membrane clamping mechanism to connect the piston and rod when pressurized. The clamping mechanism relies on friction to transmit force between the piston and the rod. This concept has been thoroughly investigated and confirmed in previous experiments. This configuration integrates two short-stroke cylinders with engaging and disengaging clamping mechanisms to function as a long-stroke actuator. Through the use of simple directional valves, the piston and rod can be securely and reliably engaged or disengaged. The sequential movement of each short-stroke piston, linked through the clamping mechanisms, produces the overall motion of the piston rod. This design enables a stepwise motion along the rod, similar to the mechanics of rope climbing actuators [5].

1.1 HILA Multi Rod

A variation known as HILA Multi-Rod (HILA MR) [6] employs multiple piston rods operating within a shared piston inside a common cylinder, see fig. 1. Each piston rod is independently controlled through separate clamping elements or static clamping members, utilizing individual logic valves. A servo valve regulates the movement of the shared piston, while independent pressurization of the clamping elements allows precise control over each linear rod. By alternating between continuous and discrete incremental positioning steps, HILA MR achieves a form of motion control that parallels the time-multiplexing schemes found in computer operating systems.

HILA MR technology replicates the functionality of hydraulic

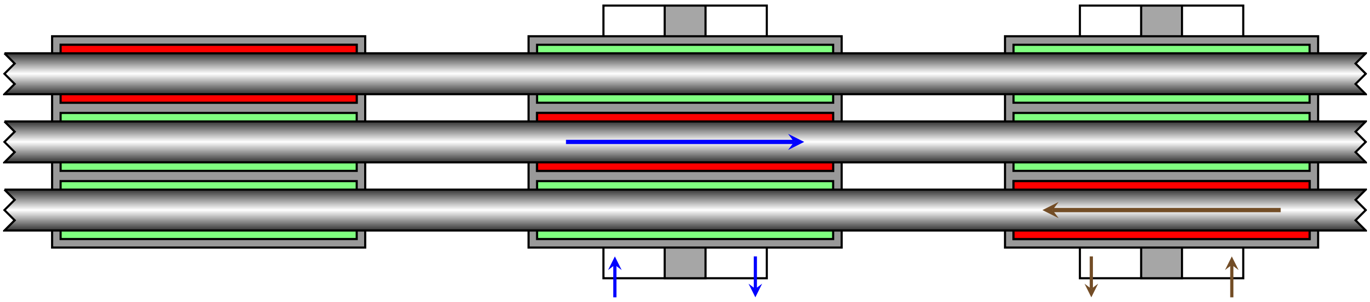

Figure 1: Multi-rod HILA with two servos controlling three rods. Red indicates that a clamp is engaged, and green disengaged. The top rod is parked, while the first servo is moving the second rod towards the right and the second servo is moving the third rod towards the left.

servo cylinders but operates by sequentially actuating multiple rods through clamping elements. Each servo piston can grip one or more rods at the same time and then move them in the desired direction. The technology enables a controlled transition between continuous motion and discrete incremental positioning, governed by precise timing. This paper investigates the special case where each piston only grips one rod at a time. This time-multiplexing approach resembles scheduling mechanisms in computer science. HILA MR technology provides a novel method for generating and distributing linear mechanical motion in various applications that require precise and scalable actuation. It bridges the gap between traditional high-fidelity hydraulic servo systems and applications with less stringent time constraints. A key advantage is its ability to enable stroke independence, reducing the force exerted on the rod while extending the stroke length without altering torque and stiffness. This flexibility enhances both design and performance, making HILA technology a promising solution for modular and efficient actuation.

In aviation, this technology offers significant advantages, including a reduced number of servo valves, the integration of well-established clamping techniques for both stepping and locking functions, and fewer position feedback sensors. A single HILA MR actuator has the potential to replace an entire actuator and sensor network for aircraft surface control. This enables substantial rationalization by reducing mass and volume per actuator function, leading to more efficient and compact aircraft control systems.

The possibilities of using HILA MR in aerospace applications were investigated by Landberg et al. [4]. The long stroke technology, combined with a high gear ratio lever arm, offers very high forces for surface control in slender wings. In this configuration, the left and middle piston rods are connected to the control cylinder. Movement and force are transmitted to the control surface through a cable or push rod and a lever arm. When applied to flight control, HILA MR shares similarities with fly-by-wire systems but differs in key aspects. The actuator is positioned within the fuselage or inside the wing box and is integrated with an electro-hydraulic power pack or linked to a redundant central hydraulic system. Rudder movements are transmitted through rigid cables or push rods that run along the wings, reminiscent of the traditional mechanical cable systems used in early aviation [7].

1.2 Clamping Mechanism

The system includes six or nine clamps, if using a single or a dual servo, respectively. Of these, three are *parking clamps*, which are attached to a fixed frame and prevents motion of the rod when engaged. The others are *transfer clamps*, which locks the rod onto a movable piston to enable transfer of motion. Figure 2 shows a simulation model of the system, where (c_{01}, c_{02}, c_{03}) are the parking clamps and $(c_{11}, c_{12}, c_{13}, c_{22}, c_{22}, c_{23})$ are the transfer clamps. The scheduling component sends control signals to each one of the clamps, as well as reference position signals to the hydraulic position servos. For simplicity, each servo is controlled by a proportional controller through closed-loop feedback.

2 Scheduling Algorithms

Four different scheduling algorithms have been implemented and analyzed in the hydraulic model; Time Division Multiplexing, Round Robin, Early-Return Round Robin, and Greedy Round Robin. They differ mainly in how they select which rod to grip and how long they wait until they grip the next rod. All algorithms are implemented in C++ and can operate on either one or two hydraulic servos. When only one servo is used, the servo must return to neutral position between each motion. If two servos are used, one servo moves the selected rod while the other performs the return motion.

2.1 Time Division Multiplexing

Time Division Multiplexing (TDM) is a technique used in telecommunications to transmit multiple signals over a single communication channel. It works by dividing the channel's available time into separate time slots and assigning each signal a unique time slot for transmission. This allows multiple data streams to share the same physical medium efficiently, avoiding interference between signals. However, TDM can conceptually be adapted as a general task scheduler, which have been studied in several fields such as controlling multiple solar panels [8], air traffic signals [9] and surgical robots [10]. The key idea of TDM - dividing available time into discrete time slices and allocating them to specific tasks - makes it a suitable framework for controlling a multi-rod HILA, where multiple servos take turns in moving the rods. The rods are selected in sequential order, and each rod is gripped for a fixed

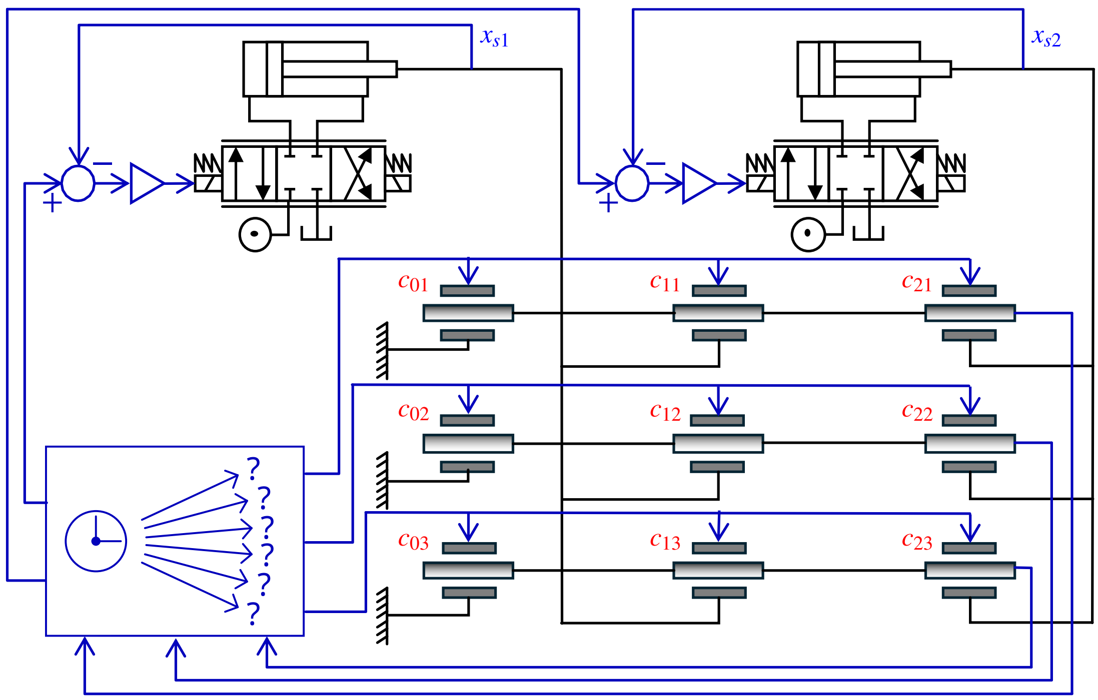

Figure 2: Schematic view of the simulation model.

period of time before it is released, a shown in figs. 3 and 4. A benefit with this approach is *predictable waiting time* - the time required for a specific motion can be calculated in advance. A critical parameter is the size of the time slots, which balances between movement resolution and overhead due to switching between rods and servos.

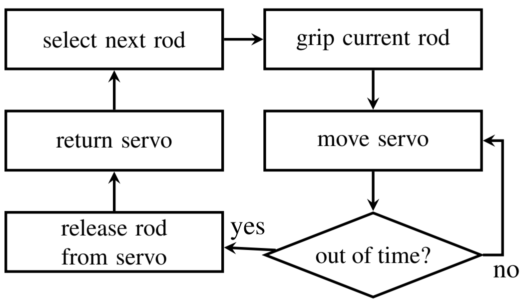

Figure 3: Flowchart of TDM algorithm for a single servo.

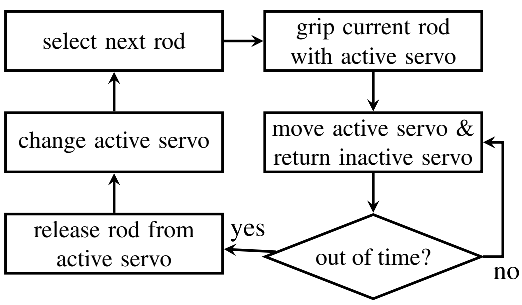

Figure 4: Flowchart of TDM algorithm for a dual servo.

2.2 Time Equidistant Round Robin

A more traditional scheduling technique is to balance the utilization by means of need and availability. Round Robin (RR) scheduling is a widely used CPU scheduling algorithm that allocates a fixed time quantum to each process in a cyclic order [11]. This approach ensures fairness by allowing all processes to execute for an equal time slice before moving onto the next. RR scheduling is particularly effective in time-sharing systems, as it minimizes starvation and provides predictable response times.

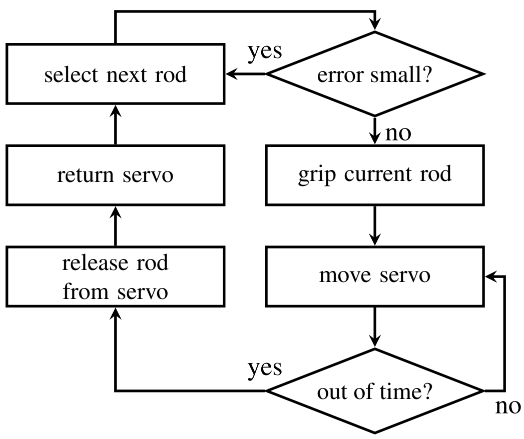

Figure 5: Flowchart of RR algorithm for a single servo.

The main difference between TDM and RR scheduling is that rods where the position error is below a given tolerance will be skipped over, so that only those rods with a significant error will be moved. Figures 5 and 6 shows flowcharts for the algorithm with one and two servos, respectively.

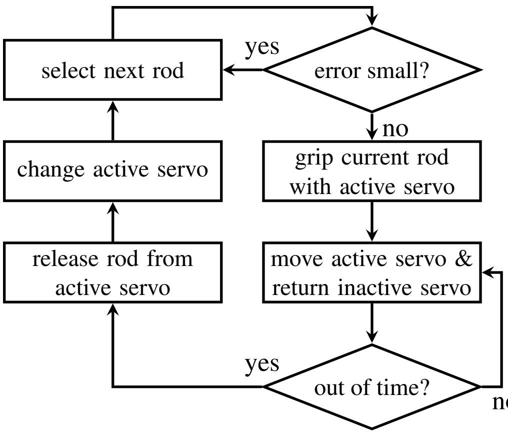

Figure 6: Flowchart of RR algorithm for a dual servo.

2.3 Early-Return Round Robin

Assigning a fixed time slice for each rod will be wasteful in case the rod reaches its reference position before the time slice has ended. The servo would then have to wait until the end of the current time slice before it is allowed to move any of the other rods. This can be avoided by aborting the motion early and changing to the next rod as soon as the position error is within a given tolerance. In this way the next rod can start moving earlier, improving the resolution of the motion, albeit at the cost of sacrificing the predictable waiting time. However, the worst-case time to complete a task can still be calculated. We denote this algorithm the *Early-Return Round Robin (ERR)* algorithm. When using two servos, it must be ensured that the second servo has returned to neutral position before the next rod can start to move. Otherwise the positions of the hydraulic servos might start to drift, eventually reaching their end-of-stroke limitations. Figures 7 and 8 shows flowcharts for the algorithm with one and two servos, respectively.

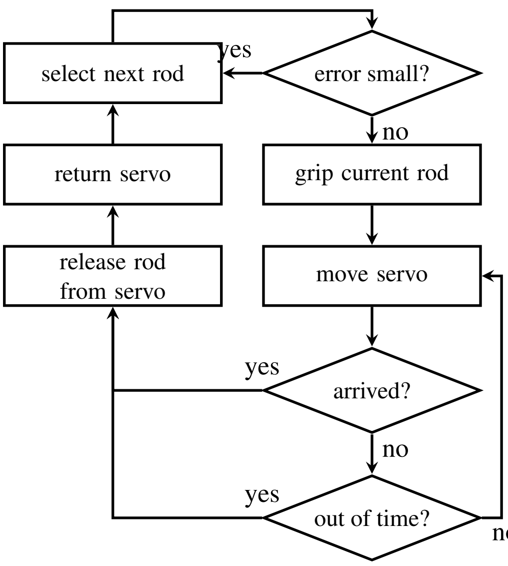

Figure 7: Flowchart of ERR algorithm for a single servo.

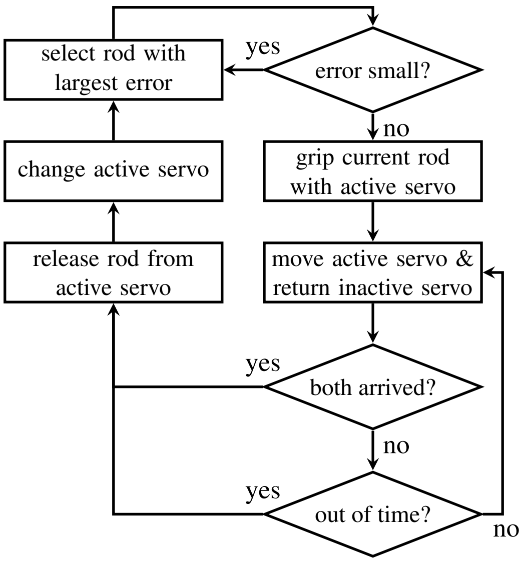

Figure 8: Flowchart of ERR algorithm for a dual servo.

2.4 Greedy Round Robin

The previous algorithms all select one rod at a time in a specific order, without priority. A possible improvement is to give priority to the rods with the largest position error. The fourth algorithm is an attempt to improve the fairness of the algorithm by adding a "greedy" rod selection. The servo always moves the rod with the largest error, calculated as the absolute difference between reference and feedback signals. Similarly, to the ERR algorithm, it also aborts the motion and selects a new rod once the current rod is close enough to its reference position. Note that this algorithm is not starvation free, and that waiting times are unpredictable. In the worst case, a rod with a small position error may need to wait very long until it is moved. Figures 9 and 10 shows flowcharts for the algorithm with one and two servos, respectively.

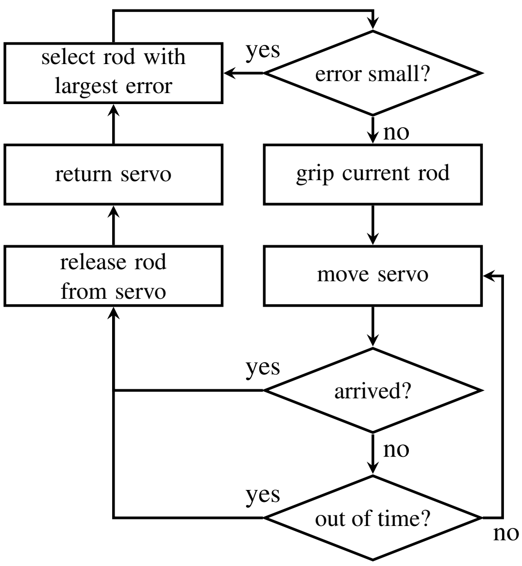

Figure 9: Flowchart of GRR algorithm for a single servo.

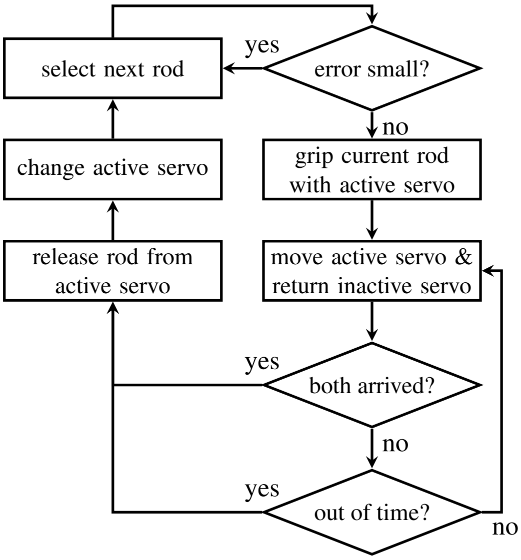

Figure 10: Flowchart of GRR algorithm for a dual servo.

3 Simulation Experiments

All algorithms are tested in the Hopsan simulation tool [12] with the model illustrated in fig. 2. Submodels for inertia and friction are left out of the figure for simplicity. It is important to evaluate the algorithms when moving one, two, or three rods simultaneously. For this reason, predefined input reference signals are used where all rods are moving, sometimes at the same time, and sometimes not. These references are used to study the deviation of the actual position of each piston rod against the corresponding input reference. The same reference input signals are used for all simulations and evaluations, as shown in fig. 11. One rod is moving all the time, one starts moving after 1 second, and the third only moves between 4 and 8 seconds. This makes it possible to compare how the different algorithms behave when moving multiple rods simultaneously. The rods also move with different frequencies. The fastest frequency is intentionally chosen high compared the time slices in the scheduling algorithm, to demonstrate the difference in dynamic response between one and two servos.

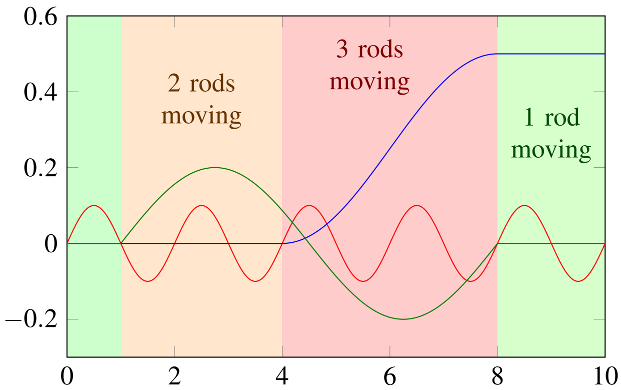

Figure 11: Reference signals for simulation experiments.

3.1 Results with Single Servo

The results for each algorithm with a single servo are shown in figs. 12 to 15. It is clear that TDM is not able to capture the fast dynamics of the first rod. The time resolution limits the maximum frequency response of the system. The other three algorithms work better, but still have problems with fast dynamics while moving all three rods at the same time. It should be noted that the time slices used are intentionally large compared to the fast dynamics of the model. In reality, one would choose time slices that are sufficiently small to capture the dynamics of the reference signals. However, the minimum size of the time slices is also limited by the response time of the hydraulic position servos.

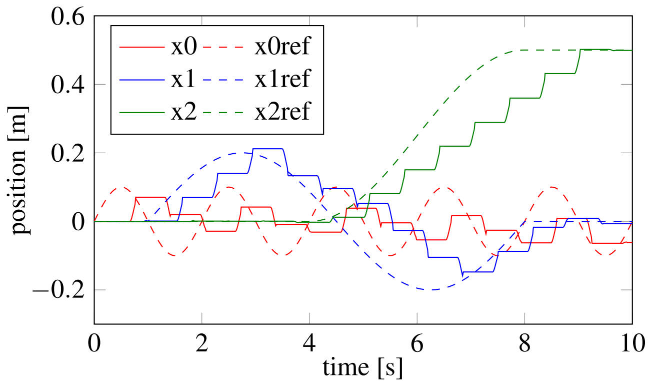

Figure 12: Simulation results for a single servo with TDM scheduling.

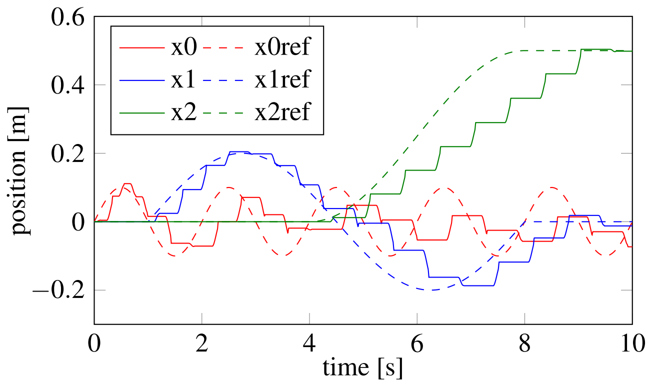

Figure 13: Simulation results for a single servo with RR scheduling.

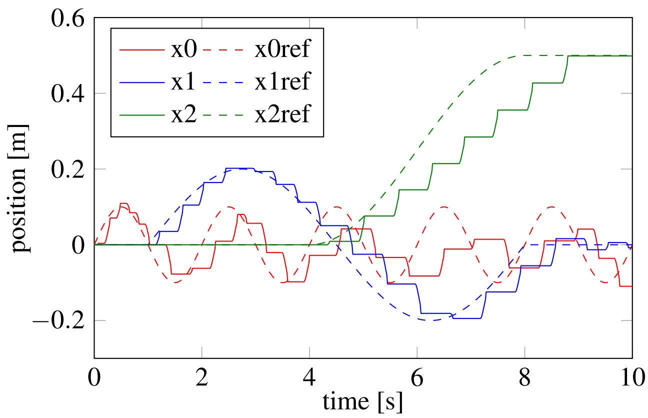

Figure 14: Simulation results for a single servo with ERR scheduling.

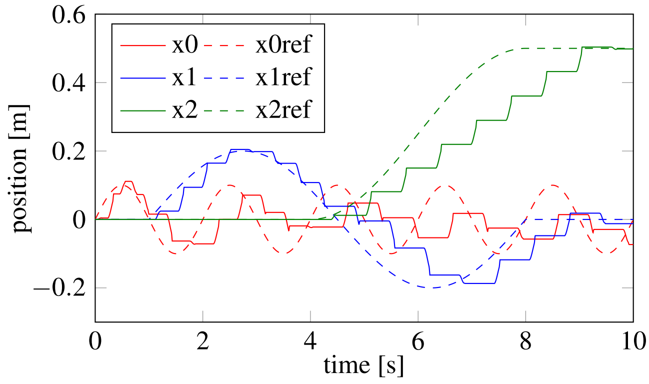

Figure 15: Simulation results for a single servo with GRR scheduling.

3.2 Results with Dual Servo

Results for a dual servo is shown in figures figs. 12 to 15 for each algorithm, respectively. All algorithms are now able to capture all frequencies in he references signals, but with different degree degrees of smoothness ranging from worst (TDM) to best (GRR). With TDM the high frequency motion suffers from a phase shift, as it is never really able to catch up with the reference signal.

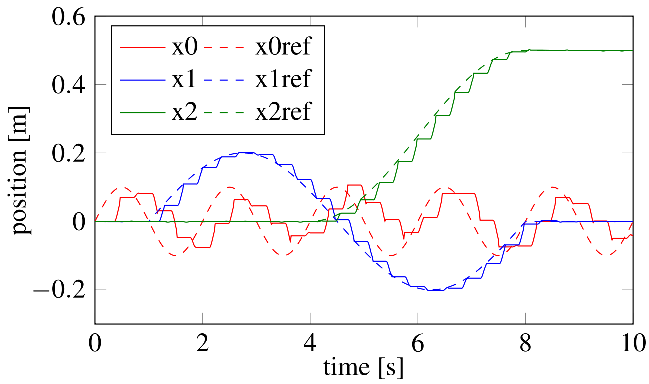

Figure 16: Simulation results for a dual servo with TDM scheduling.

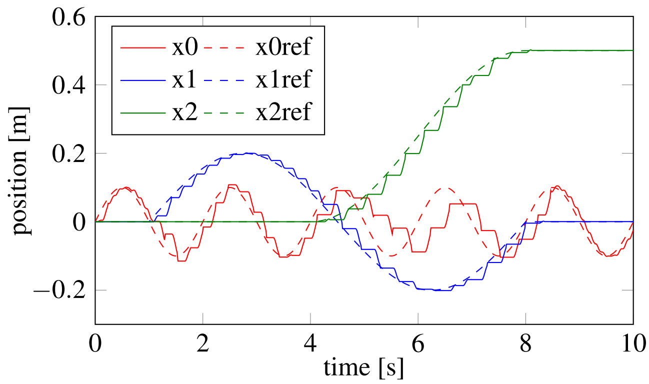

Figure 17: Simulation results for a dual servo with RR scheduling.

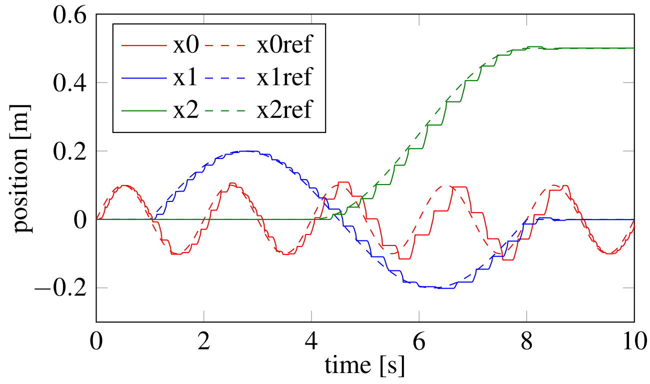

Figure 18: Simulation results for a dual servo with ERR scheduling.

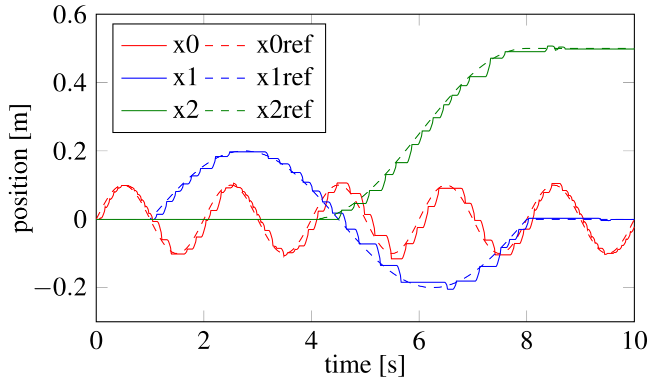

Figure 19: Simulation results for a dual servo with GRR scheduling.

3.3 Prioritization of Rods

In safety critical systems, not least aerospace applications, some motions are more important than others. For example, the control surfaces on an airplane is more important than the flaps and the landing gear. Hence, the scheduler must be able to give priority to these motions. This feature was added to the GRR algorithm. If a prioritized rod needs to move, no other rod is allowed to move until the prioritized rod has reached its reference position. Results with a single and a dual servo is shown in figs. 20 and 21, respectively.

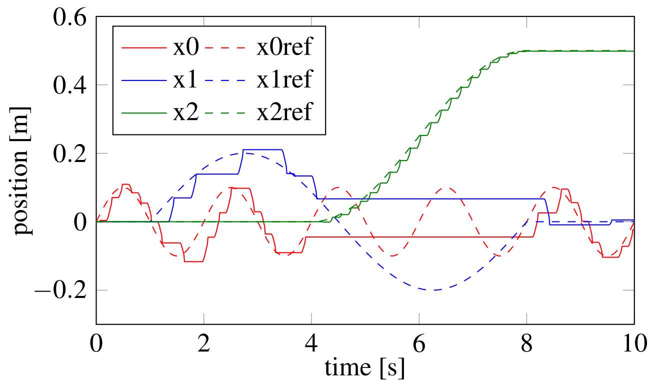

Figure 20: Simulation results for a single servo with GRR scheduling and priority for x2.

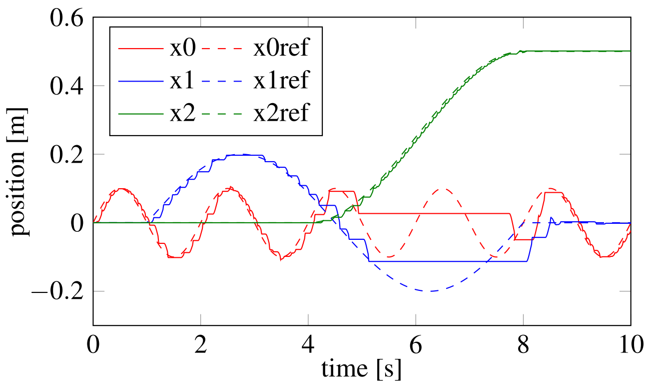

Figure 21: Simulation results for a dual servo with GRR scheduling and priority for x2.

algorithm	single servo	dual servo
TDM	203 mm	136 mm
RR	205 mm	130 mm
ERR	188 mm	97 mm
GRR	154 mm	48 mm

Table 1: Maximum error from any one of the rods during the reference cycle.

algorithm	single servo	dual servo
TDM	177 mm	77 mm
RR	155 mm	55 mm
ERR	132 mm	39 mm
GRR	103 mm	32 mm

Table 2: Average total error during the reference cycle.

4 Evaluation

The simulation results indicate that time-sharing scheduling algorithms, such as TDM and RR, can be effectively applied to HILA MR systems, both with one and two servos. From a scheduling perspective, it would be straightforward to extend the system to incorporate a third servo or additional rods. However, it is important to acknowledge that the simulation model contains several unknown model parameters. These include both design-related variables, depending on component selection, and physical effects, such as friction and leakage, which could influence the accuracy and predictive capability of the model.

The addition of a second servo significantly improves the dynamic response of the system. However, this enhancement comes at the cost of increased complexity, weight, size, and cost. Furthermore, the dynamic behavior of the system is highly dependent on the time slices allocated by the scheduling algorithm. If the time slices are sufficiently short relative to the fastest dynamics in the system, even a single-servo configuration could adequately capture the required motion characteristics.

Among the evaluated algorithms, GRR demonstrated the best overall performance. However, it is not starvation-free, meaning that all rods cannot be guaranteed to receive actuation within finite time. Although this issue is unlikely to occur in practice, it poses a potential risk in safety-critical applications. To mitigate this limitation, we have shown that GRR can be enhanced with a priority mechanism, ensuring that safety-critical motions are always executed when necessary.

To quantitatively compare the algorithms, maximum and average errors were extracted from the simulation data. Table 1 presents the maximum positional error observed for any individual rod across different algorithm and servo configurations, while Table 2 reports the average total error accumulated across all rods during a full cycle. Both tables further confirm that GRR consistently outperforms the other algorithms in terms of accuracy and control precision.

5 Conclusions

The HILA MR approach enables the design and control of more compact actuation systems, as a single piston or a small number of pistons can control multiple piston rods. In addition, the stroke of each cylinder is not constrained by the stroke of the rods, allowing short cylinders to perform long-range motions. This characteristic is particularly valuable in applications where weight and volume are critical constraints, such as aerospace engineering.

This study demonstrates the feasibility of controlling HILA MR systems using various time-sharing scheduling algorithms. Moreover, it highlights the possibilities of assigning priority to specific rods, an essential feature for safety-critical applications. Among the tested algorithms, Early-Return Round Robin and Greedy Round Robin exhibited the most promising performance.

Future research directions include experimental validation of these algorithms on a physical hardware system, as well as the integration of more advanced and realistic models into the simulation framework. In addition, exploring more sophisticated scheduling strategies, such as algorithms that allow a single piston to grip multiple rods simultaneously, could further enhance system efficiency and performance.

References

[1] Magnus Landberg, Martin Hochwallner, and Petter Krus. Novel linear hydraulic actuator. In *Fluid Power Systems Technology*, volume 57236, page V001T01A054. American Society of Mechanical Engineers, 2015.

[2] Magnus Landberg, Magnus Sethson, and Petter Krus. The hydraulic infinite linear actuator for efficient and flexible timber and agricultural logistics. In *Scandinavian International Conference on Fluid Power*, pages 309–321, 2021.

[3] Magnus Landberg, Robert Braun, Magnus Sethson, and Petter Krus. The hydraulic infinite linear actuator applied to elevators in mid-rise buildings. In *Fluid Power Systems Technology*, volume 51968, page V001T01A030. American Society of Mechanical Engineers, 2018.

[4] Magnus Landberg, Magnus Sethson, Robert Braun, and Petter Krus. An actuator system to control multiple surfaces in a morphing wing. In *:*, pages 8–, 2023.

[5] Martin Hochwallner, Magnus Landberg, and Petter Krus. The hydraulic infinite linear actuator: Properties relevant for control. In *10th International Fluid Power Conference (10. IFK, Dresden, Germany*, volume 3, pages 411–424, 2016.

[6] Magnus Landberg, Magnus Sethson, and Petter Krus. The hydraulic infinite linear actuator with multiple rods. In *The 15th Scandinavian International Conference on Fluid Power, SICFP'17, June 7-9, 2017, Linköping, Sweden*, volume 144, pages 279–293. Linköping University Electronic Press, 2017.

[7] Martin Wehr, Andreas Pott, and Karl-Heinz Wehking. Bending fatigue strength and lifetime of fiber ropes. In Clément Gosselin, Philippe Cardou, Tobias Bruckmann, and Andreas Pott, editors, *Cable-Driven Parallel Robots*, pages 73–84, Cham, 2018. Springer International Publishing.

[8] Oscar López-Lapeña. Time-division multiplexing control of multi-input converters for low-power solar energy harvesters. *IEEE Transactions on Industrial Electronics*, 65(12):9668–9676, 2018.

[9] Ben Ahmed Mohamed, Boudhir Anouar Abdelhakim, Bouhorma Mohammed, and Ben Ahmed Kaoutar. A statistical multiplexing method for traffic signal timing optimization in smart cities. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, 15(1):107–113, 2015.

[10] Yashuai Zuo, Glen Merritt, and Xuefeng Wang. Design of a novel surgical robot with rigidity-adjustable joints based on time-division multiplexing actuation. In *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 885–890. IEEE, 2020.

[11] Taghreed Balharith and Fahd Alhaidari. Round robin scheduling algorithm in cpu and cloud computing: a review. In *2019 2nd international conference on computer applications & information security (ICCAIS)*, pages 1–7. IEEE, 2019.

[12] Robert Braun, Peter Nordin, Liselott Ericson, L. Viktor Larsson, Petter Krus, and Maria Pettersson. Hopsan: An open-source tool for rapid modelling and simulation of fluid and mechatronic systems. In *Proceedings of the BATH/ASME 2020 Symposium on Fluid Power and Motion Control (FPMC2020) :*. American Society of Mechanical Engineers (ASME), 2020.