Control allocation for an industrial high-precision transportation and positioning system*

R. Beerens\textsuperscript{1}, S.C.N. Thissen\textsuperscript{2}, W.C.M. Pancras\textsuperscript{3}, T.M.P. Gommans\textsuperscript{4}, N. van de Wouw\textsuperscript{1,5}, W.P.M.H. Heemels\textsuperscript{1} Fellow, IEEE

Abstract—We present a control allocation framework to improve the performance of an industrial high-precision translational transportation and positioning system, based on an inverted permanent-magnet linear synchronous motor. Compared to the state-of-practice control solution, the proposed allocation technique achieves enhanced tracking performance, enlarged motion freedom, minimizes power consumption, and realizes relaxed hardware design specifications. A decentralized allocation algorithm is presented that enhances computational efficiency, which facilitates the scalability to larger system configurations. The performance benefits of the proposed technique are illustrated by means of an experimental case study.

Index Terms—Motion control, control allocation, inverted permanent-magnet synchronous motor

I. INTRODUCTION

In this paper, we present a control allocation framework to improve the performance of industrial high-precision transportation and positioning systems (see Fig. 1), with respect to current state-of-practice control solutions. In particular, we consider a linear (i.e., translational) motion system, where multiple carriers can move on horizontal tracks, using the inverted permanent-magnet linear synchronous motor (IPMLSM) actuation principle, see, e.g., [1]. In such a system, the coils of the three-phase actuators are located on the tracks and the permanent magnets are placed on the carriers, see Fig. 1, such that the moving carriers do not have any electronics or cables attached to them. This principle makes the system highly suitable to be used in automated production lines involving, e.g., operation in vacuum, high temperature, or chemical environments. Due to these characteristics, such systems are widely used in industry in, e.g., the production of flat screens, OLED lighting, and photovoltaic solar cells [2].

Multiple carriers, transporting (semi-finished) products, can move over fixed guidance tracks of arbitrary length in an automated production line. At specific locations along the production line, the carriers may be required to be positioned accurately, such that a certain operation on the product can be executed. At the same time, a subsequent carrier may move towards this operation stage and queues until the first carrier has moved on. This requires the system to allow for flexible, and independent, motion of each carrier on arbitrary positions on the track. Furthermore, the system is typically over-actuated, since a carrier may commute with more than one set of coils in the track at the same time (i.e., one carrier is actuated by multiple actuators simultaneously), or multiple carriers may be influenced by the same set of coils (actuator) simultaneously. This may lead to conflicting control objectives for the actuators, and results in large position errors or the inability to control carriers independently when using state-of-practice control solutions. Moreover, the actuator characteristics suffer from position dependency and end-effects (to be discussed in more detail in Section II), which pose design limitations in the sense that actuators must currently be placed at specific locations on the tracks. In this work, we aim to overcome the above limitations by means of intelligent (allocation-based) control.

IPMLSM-based transportation systems are often controlled using Field Oriented Control (or vector control), see, e.g., [3], [4], where actuator redundancy is often handled through the use of commutation algorithms, see [1], [5]. However, these existing commutation algorithms cannot handle independent control of multiple carriers simultaneously. Other control techniques that can handle actuator redundancy are, e.g., optimal control (see, e.g., [6], [7]) or control allocation (see, e.g., [7], [8]). Well-known examples of the former are linear quadratic control [6, Ch. 15], $H_\infty$ control [6, Ch. 16-18], or model predictive control (MPC) [9]. For the current application, however, we choose to apply control allocation, since it offers several benefits over optimal control: it invokes a lower computational effort [7], and (most importantly) separates
controller design and tuning from the distribution of the resulting control efforts. In this way, well-known loop-shaping techniques (see, e.g., [10]), often used in industry for the tuning of motion controllers, can still be applied. Control allocation has been applied on a wide range of applications, e.g., on over-actuated electric vehicles [11], and harddisk drives [12].

The main contributions of this paper are as follows. The first contribution is a control allocation framework for an IPMLSM-based transportation and positioning system. Compared to the state-of-practice control solution currently applied by the manufacturer, the proposed allocation technique achieves 1) improved tracking performance, 2) the possibility to perform independent motion of multiple carriers, 3) minimization of power consumption, 4) a relaxation of hardware design rules, and 5) the possibility to take actuator limits into account. Second, we propose a decentralized control allocation approach that allows for real-time implementation of the proposed allocation scheme, by significantly reducing computational effort. The third contribution is an experimental case study on an industrial application. This paper builds upon our previous work in [13], which contains the controller design and a simulation example. In addition to [13], this paper contains a novel decentralized control allocation approach, and an extensive experimental study.

The remainder of this paper is organized as follows. In Section II, we provide a detailed system description, and the proposed control allocation architecture is discussed in Section III. We illustrate the achievable performance benefits of the proposed controller by an experimental study in Section IV, and provide conclusions in Section V.

II. SYSTEM DESCRIPTION

This section starts with the basic actuation principle of an IPMLSM and the control challenges, followed by a model of the considered transportation system.

A. Actuation principle and control problems

An inverted linear permanent-magnet synchronous motor has a magnet array attached to the carriers, and coils fixed to the tracks. In PMSM-based motion systems, the required three-phase currents to obtain the desired motion profile are typically generated in the so-called dqθ-reference frame (see, e.g., [4, Ch. 6], [14, Ch. 10]), to simplify the control problem. Consider Fig. 2. The three-phase currents $i_a$, $i_b$, and $i_c$ (red) can be mapped onto the stationary $\alpha_1, \beta_1$-frame (blue) via the Clarke transformation [14, Ch. 10]. Next, the coil currents expressed in the dqθ-frame are obtained by the Park transformation [14, Ch. 10], i.e., by rotating the $\alpha_1, \beta_1$-frame by the commutation angle $\theta$. The coil currents are now expressed by the direct current $i_d$ and quadrature current $i_q$ (green). For a translational system, this transformation results in a moving reference frame fixed to the carrier, where the $q$-direction is aligned with the direction of movement, by choosing the commutation angle in a specific way (depending on the carrier position). In this way, it can be realized that $i_q$ is the only force-generating current by controlling $i_d$ to zero, such that only one input signal needs to be generated by a motion controller to achieve the desired motion of the carriers.

Fig. 2. Current reference frames for PMSMs.

The considered inverted linear PMSM, however, suffers from end-effects. Due to the fact that the stator is segmented into groups of three-phase coils, there exist regions where the electronics in the tracks partially overlap a magnet array on a carrier. The correct commutation angle (i.e., such that $i_q$ indeed implements the desired control force, generated by the motion controller, on the carrier) is then a nonlinear function of the carrier position. The motor gain (i.e., the gain between the applied quadrature current and resulting force on the carrier) thus depends on the carrier position and the commutation angle.

The segmentation of coils also gives rise to the following control problems:

1) Given typical physical dimensions, carriers may be influenced by either one or two sets of coils (from now on referred to as “actuators”), leading to an over-actuated system.

2) Actuators may also influence multiple carriers at the same time. However, since an actuator is only able to implement a correct commutation angle (and thus a correct control force) for a single carrier, the other carrier experiences large disturbance forces as a result of the difference between the desired and attained control force. This restricts the freedom in motion profile design, in the sense that two adjacent carriers cannot perform an independent motion accurately, as we will illustrate in Section IV.

3) Besides the aforementioned motion restrictions, an inverted linear PMSM suffers from hardware restrictions. The spacing between actuators has to be specifically related to the length of the magnet array on the carrier to achieve a smooth transition of a carrier between two actuators, see Fig. 3 (where we assume that the magnet arrays have the same length as the carriers). That is, the center distance of two actuators must be equal to the length of the magnet array $L$. When using this particular spacing, the sum of the motor gains of the actuators that influence a given carrier is constant for all carrier positions. Then, if the same quadrature current $i_q$ is applied to multiple actuators, the applied force is independent of the carrier position. That is, there is no difference in the
applied force to a carrier, regardless of whether the carrier is influenced by either one or two actuators. In this way, a single carrier may indeed be controlled accurately when actuated by multiple actuators (but does not allow for the control of multiple neighbouring carriers by the same actuator).

To address the above limitations, we propose a control allocation framework in Section III that results in 1) enhanced tracking performance, 2) allowing independent motion of multiple carriers, despite the conflicting control objectives, 3) reduces power consumption, 4) relaxed hardware design specifications, and 5) the possibility to take actuator limitations into account.

B. Carrier transportation system modeling

Consider an IPMLSM-based carrier transportation system consisting of \( n \) carriers and \( m \) actuators, see Fig. 3. Let \( j \in n := \{ 1, 2, \ldots, n \} \) be the carrier number, and \( k \in m := \{ 1, 2, \ldots, m \} \) the actuator number, used to uniquely identify all carriers and actuators in the system. The carrier transportation system is governed by the following dynamics:

\[
M \ddot{y} = B(y)u,
\]

with \( y = [y_1, \ldots, y_n]^\top \) the position of the carriers (\( y_j = 0 \) indicates the lower end of the tracks), and \( M \) the diagonal mass matrix containing the carrier masses. We consider the \( \alpha, \beta \)-currents for each actuator as control inputs, instead of the frequently used \( i_q \)-current of the dq0-frame. The essential motivation of using the fixed \( \alpha, \beta \)-frame is that the motor gain matrix \( B(y) \) in (3) below is then independent of the commutation angle \( \theta \). The motor gain matrix \( B(y) \) hence solely depends on the carrier positions, thereby simplifying the allocation problem in Section III. The input vector \( u \) is then given by

\[
u = [i_{\alpha 1}, i_{\beta 1}, \ldots, i_{\alpha m}, i_{\beta m}]^\top. \tag{2}\]

Due to the position dependency of the commutation between the carriers and the coils, the motor gain matrix is given by

\[
B(y) = \begin{bmatrix}
 b_{11}^\alpha (y_1) & b_{11}^\beta (y_1) & \ldots & b_{1m}^\alpha (y_1) & b_{1m}^\beta (y_1) \\
 \vdots & \vdots & \ddots & \vdots & \vdots \\
 b_{m1}^\alpha (y_n) & b_{m1}^\beta (y_n) & \ldots & b_{mn}^\alpha (y_n) & b_{mn}^\beta (y_n)
\end{bmatrix}, \tag{3}
\]

where \( b_{jk}^\alpha \) and \( b_{jk}^\beta \) are actuator-specific, position-dependent motor gains. The right-hand side of (1) then results in a column with forces applied on the carriers, i.e.,

\[
\tau = B(y)u = \left[ \sum_{k=1}^m \tau_{1k} \ldots \sum_{k=1}^m \tau_{nk} \right]^\top. \tag{4}
\]

In (4), \( \tau_{jk} \) is the force applied by actuator \( k \) on carrier \( j \), and is straightforwardly given by

\[
\tau_{jk} = b_{jk}^\alpha (y_j) i_{\alpha k} + b_{jk}^\beta (y_j) i_{\beta k}. \tag{5}
\]

Note that some elements \( \tau_{jk} \) in (4) may be zero if actuator \( k \) does not influence carrier \( j \), which is the case when the carrier is not close enough to the actuator in order to commute.

Let us introduce the relative position of carrier \( j \) with respect to actuator \( k \), i.e., \( z_{jk} := y_j - Y_k \), with \( Y_k \) the position of actuator \( k \) on the tracks, defined as the minimum position \( y_j \) where actuator \( k \) starts influencing a carrier \( j \), i.e., \( Y_k := \min \left\{ y_j \mid b_{jk}^\alpha (y_j) \neq 0 \lor b_{jk}^\beta (y_j) \neq 0 \right\} \), which is independent of \( j \). We are now ready to pose the following assumptions regarding controllability of each carrier and similarity of the hardware components.

Assumption 1. \( \text{Rank}(B(y_j)) = n \).

Assumption 2. All three-phase coil segments and permanent magnet arrays are identical.

As a result of Assumption 1, any carrier at any position on the tracks is influenced by at least one actuator, so that any desired \( \tau_{jk} \) can be realized by the actuators. Assumption 2 is not necessarily needed for the developments in this paper, but leads to significantly reduced complexity of \( B \) and therefore easier implementation. A consequence of this assumption is that the motor gains of each actuator \( k \) with respect to each carrier \( j \) are identical. We can then simplify the motor gain matrix \( B \) in (3) by writing \( b_{jk}^\alpha \) and \( b_{jk}^\beta \) in (3) as:

\[
b_{jk}^\alpha (y_j) = b_\alpha (z_{jk}), \quad b_{jk}^\beta (y_j) = b_\beta (z_{jk}). \tag{6}
\]

The motor gains in (6) are typically obtained from FEM-based electromagnetic simulations on the interaction between a carrier and an actuator, see [1]. By Assumption 2, we only have to perform these simulations for a single actuator/cARRIER interaction, thereby simplifying the implementation of the control allocation architecture below (where the gain matrix \( B \) is explicitly used).

III. CONTROL ARCHITECTURE

We will first discuss the high-level control scheme, and subsequently the allocation algorithm. Finally, we present a decomposition algorithm that enhances computational efficiency, so that the control allocation architecture can be evaluated online at high sampling rates and facilitates implementation on large-scale system configurations.

A. High-level control scheme

Consider Fig. 4. The reference signals \( r \) (one for each carrier on the track) and error signals \( e \) between the reference and the carrier positions \( y \) are provided to a motion controller (composed of a feedback and feedforward part). This controller is typically designed using well-known loop-shaping techniques, and generates the desired control forces \( \tau_c = [\tau_{c1}, \ldots, \tau_{cn}]^\top \). These desired control forces should then be applied to the carriers by the AC actuators. Assuming that the motion controller is stabilizing by proper design, closed-loop stability is then preserved through control allocation. The primary objective of the control allocator is thus to find

\[
\tau \in \mathbb{R}^{n\times1}
\]

\[
\tau_{fb} \in \mathbb{R}^{n\times1}
\]

\[
\tau_{ff} \in \mathbb{R}^{n\times1}
\]

\[
\text{controller allocation}\quad \text{actuator electronics}\quad \text{plant}
\]

\[
\tau_{fb} \quad \tau_{ff}
\]

\[
\text{C}_{fb} \quad \text{C}_{ff}
\]

Fig. 4. High-level control architecture. \( C_{fb} \) and \( C_{ff} \) indicate the feedback and feedforward parts of the motion controller, respectively. The superscripts on the signals indicate their dimensions.
the currents $i_{\alpha k}$ and $i_{\beta k}$ in the control input $u$ in (2), such that
the actuators indeed implement the desired control forces given by $\tau_c$. In other words,
the actual forces acting on the carriers, denoted by $\tau$ in Fig. 4, should be equal to the desired
control forces as determined by the motion controller, i.e., $\tau = B(y)u = \tilde{\tau}_c$. Due to the over-actuated
nature of the system, the solution to the allocation problem $\tau = \tau_c$ (if attainable by the actuators) is not unique. We will exploit this
freedom to introduce a second control objective, namely the
minimization of power consumption by the actuators.

We will now discuss an unconstrained and a constrained
allocation procedure. The latter incorporates a saturation con-
straint on the input, which is motivated by the desire from
industry to use cost-effective (less powerful) actuators.

### B. Control allocation

We aim at implementing the desired control force generated
by the motion controller on each carrier, while minimizing
the power consumption. The control allocation can then be
described by the following convex quadratic programming problem (QP) [15, Ch. 4]:

$$\begin{align*}
\min_{u} & \quad u^TWu, \\
\text{subject to} & \quad \tau_c = B(y)u,
\end{align*}$$

(7)

where $u^TWu$ is a quadratic measure for the power con-
sumption, and $W$ is a symmetric weighting matrix. Since the
optimization problem in (7) is convex and only contains one
equality constraint, the explicit solution is given by [16]:

$$u = W^{-1}B^T(y)\left(B(y)W^{-1}B^T(y)\right)^{-1}\tau_c. \tag{8}$$

In case there is a maximum allowable control input due to,
e.g., cost-effective hardware design, the criterion $\tau = \tau_c$ may
not be attainable at all times. To deal with this scenario, we
introduce a constraint on $u$ in the form of a limit on the current,
and the difference between the desired, and attained control
forces $\epsilon_u = \tau_c - \tau$. This is then combined in an allocation
problem that can be described by the convex, quadratically
constrained quadratic program (QCQP) [15, Ch. 4] given by

$$\begin{align*}
\min_{u,\epsilon_u} & \quad \epsilon_u^TQ\epsilon_u + u^TWu, \\
\text{subject to} & \quad \epsilon_u = \tau_c - B(y)u, \\
& \quad \epsilon_u^2 + i_{\alpha k}^2 + i_{\beta k}^2 \leq i_{\text{max},k}^2, \quad \forall k \in \bar{m},
\end{align*}$$

(9a)

(9b)

with $W$ and $Q$ symmetric weighting matrices, and $i_{\text{max},k}$ is the current
limit on actuator $k$. The QCQP in (9) can be
solved online using efficient algorithms as CPLEX [17] or
Gurobi [18]. However, we propose a specific decomposition of
the optimization problem in (9) to enhance computational
efficiency. This favors solving (9) online at high sampling rates in large scale industrial applications with many carriers and actuators. The proposed decomposition method also favors
online computations of (8) in the unconstrained case.

**Remark 1.** Although the constraint in (9b) indeed resembles
a constraint on the maximum current to be provided by
the actuator, we can approximate the constraint by a set of
polyhedral constraints. Then, the control problem reduces to
a linear allocation problem [8], for which efficient quadratic
programming (QP) algorithms are available, e.g., active-set or
interior-point methods [19], which are studied in the context of
control allocation in, e.g., [20] and [21].

**Remark 2.** We choose here to perform the control allocation in
the $\alpha, \beta$-frame. In this way, we do not have to find an optimal
commutation angle $\theta$ for each actuator, which would appear in
the above minimization problems as an extra decision variable
when we would have chosen to apply the allocation in the $dq0$-
frame. Moreover, the motor gain matrix $B$ in (3) would then
not only depend on the carrier position $y$, but on $\theta$ as well,
making the equality constraint in (7) nonlinear. Performing
the allocation in the $\alpha, \beta$-frame thus significantly simplifies
the control allocation problem.

### C. Distributed control allocation

To reduce computational costs (which favors real-time
implementation on large-scale transportation systems), we
decompose the allocation problems in (7) and (9), by 1)
excluding actuators, which do not influence any carrier, from
the allocation problem, and 2) solving multiple, less complex
allocation problems subsequently, instead of the full allocation
problem at once. The decomposition is done, at every sampling
instant, by assigning carriers and actuators to specific subsets, and
solving the optimization problem (7) or (9) for each subset separately.
The only requirement for the proposed decomposition to apply is that the matrices $Q$ and $W$ are diagonal (to have decoupled costs in (7) and (9)). By the dimensions of the carriers, each carrier is influenced by up to
two actuators, whereas each actuator can influence up to two
 carriers. We first introduce the boolean variables $c_{jk} \in \{0, 1\}$, given by

$$c_{jk} = \begin{cases} 1, & \text{if } b_\alpha(z_{jk}) \neq 0 \lor b_\beta(z_{jk}) \neq 0, \\
0, & \text{otherwise,}
\end{cases} \tag{10}$$

for any $j \in \bar{n}$ and $k \in \bar{m}$. In other words, $c_{jk} = 1$ holds if
and only if actuator $k$ influences carrier $j$.

Let us now discuss assigning the subsets of carriers. A carrier
subset is given by $J_p \subseteq \bar{n}$, $p \in \bar{p} := \{1, 2, \ldots, P\}$ (where
$p$ is the subset number, and $P$ the total number of subsets), and
the related actuator subset is given by $K_p \subseteq \bar{m}$. Explaining
the assignment of carriers and actuators to the subsets is best
done by considering the problem in terms of a bipartite graph
(see, e.g., [22, Ch. 5]). Consider hereto Fig. 5. Let $G = (V, E)$
be a bipartite graph with $m+n$ vertices, consisting of $n$ carrier
vertices $C_1, \ldots, C_n$, and $m$ actuator vertices $A_1, \ldots, A_m$ (see Fig. 5). Hence, $V = \{C_1, \ldots, C_n, A_1, \ldots, A_m\}$. The set of
dges $E \subseteq V \times V$ is such that there is an edge between
carrier vertex $C_j$ and actuator vertex $A_k$ if and only if $c_{jk} = 1$
(cf. (10)). To explain our procedure to arrive at $J_p$ and $K_p$, $p \in \bar{p}$, we need the following definitions. A path (of length $l$) is
given by $v_1v_2 \ldots v_l$, where $(v_i, v_{i+1}) \in E$, $i = 1, \ldots, l-1$
in (our case, this implies that a path consists of alternating
carrier and actuator vertices). A subgraph $G' = (V', E')$ of
$G$ is a graph $V' \subset V$ and $E' \subset E$, where $E'$ only consist of
dges $(v, v')$ with both $v$ and $v' \in E'$. A (sub)graph is called
connected if there is a path between all the vertices of the
(sub)graph. With this terminology in place, we can now state the decomposition of the allocation problems (7) and (9).

In our problem, we decompose the set \( V \) now into its connected subgraphs with vertex sets \( V_p, p' \in \bar{p} := \{1, \ldots, P'\} \) (for which algorithms exist, see, e.g., [23, Ch. 6]). In case there are sets \( V_p, p' \in \bar{p} \), consisting only of actuators, we discard these sets such that we can exclude the associated actuators from the allocation problem, because these do not influence any carrier (i.e., the currents \( i_a \) and \( i_b \) are zero for those particular actuators). A reordering then leads to the sets \( V_p \) with vertex sets \( \bar{p} \in \bar{p} \), where \( P \leq P' \) because the sets in \( V_{p'} \) consisting only of actuator vertices are discarded, see Fig. 5. Due to physical properties, there are no sets in \( V_1, \ldots, V_P \) that only consists of carrier vertices, as each carrier is influenced by at least one actuator, see Assumption 1.

We decompose the remaining sets \( V_p, p \in \bar{p}, V_p = J_p \cup K_p \), with \( J_p \) consisting only of carrier vertices and \( K_p \) consisting only of actuator vertices, as indicated in Fig. 5. In this way, we obtain that the carriers are assigned to disjoint subsets \( J_p \subseteq \bar{m}, p \in \bar{p} \), where \( J_p \) contains the indices of the carriers contained in \( J_p \), i.e., \( J_p = \{C_j \mid j \in J_p\} \). Similarly, the actuators are assigned to disjoint subsets \( K_p \subseteq \bar{m}, \), \( p = 1, 2, \ldots, P \). Clearly, \( J_p \) and \( K_p \) are related in the sense that the carriers in \( J_p \) are only influenced by actuators in \( K_p \) (and not by any others, see Fig. 3 and 5). Moreover, an actuator in \( K_p \) has an influence on at least one carrier in \( J_p \), i.e., if \( k \in K_p \) there is \( j \in J_p \) such that \( c_{jk} = 1 \). We have that \( J_p \cap J_{p'} = \emptyset \) when \( p \neq p' \) and \( \bigcup_{p=1}^{P} J_p = \bar{m} \) by Assumption 1 (i.e., \( \{J_1, \ldots, J_P\} \) forms a partition of \( \bar{m} \)). Also, \( K_p \cap K_{p'} = \emptyset \) when \( p \neq p' \) and \( \bigcup_{p=1}^{P} K_p < \bar{m} \), and thus \( \{K_1, \ldots, K_P\} \) forms a partition of \( \{k \in \bar{m} \mid c_{jk} = 1 \} \). Hence, actuators that do not influence any carrier are not in \( \bigcup_{p=1}^{P} K_p \). As a result, these actuators are excluded from the allocation problem, improving computational efficiency. Note that two carriers are in the same set \( J_p \) if and only if there is a connection (path) via multiple actuators/carriers between them (cf. Fig. 5) and, hence, in the allocation problem these have to be treated simultaneously. Sets \( J_p \) (combined with \( K_p \)) and \( J_{p'} \) (combined with \( K_{p'} \)) for \( p \neq p' \) can be considered separately in the allocation problem as there is no path between any of the elements in \( J_p \) (and \( K_p \)) and \( J_{p'} \) (and \( K_{p'} \)).

Algorithms are available to decompose a graph in its connected subgraphs (see, e.g., [23, Ch. 6]). However, in our case, we obtain the sets \( J_p \) and \( K_p \), \( p \in \bar{p} \) by exploiting a physical ordering of the carriers and actuators (without loss of generality): the carriers and actuators are logically numbered 1 to \( n \) from left to right on the tracks (see also Fig. 3). Moreover, using the fact that a carrier is influenced by either one or two carriers, and each actuator influences at most two carriers, we can simplify the generic algorithms typically used to decompose a graph into its connected subgraphs (see, e.g., [23, Ch. 6]), leading to Algorithms 1 and 2 below. We use these algorithms in the experimental results that follow later in Section IV.

As a result of the decomposition, the solution for \( u \) in (8) in the unconstrained allocation problem (7) changes to the following set of solutions:

\[
\begin{align*}
\min_{u^p, e^p} & \quad e^p_T Q_p e^p + u^p_T W_p u^p \\
\text{subject to} & \quad e^p = \tau^p - B_p(y^p)u^p, \\
& \quad i_{ak}^2 + i_{bk}^2 \leq i_{\text{max},k}^2, \quad \text{for all } k \in K_p,
\end{align*}
\]

for all \( p \in \bar{p} \), where \( y^p \) and \( \tau^p \) are the positions of the carriers and the desired control forces in subset \( J_p \), respectively. Moreover, \( B_p \) and \( W_p \) correspond to the submatrices of the motor gain matrix \( B \) and the (diagonal) weighting matrix \( W \), respectively, corresponding to \( K_p \). Using this decomposition method, the computational effort (in the sense of computing the inverse in (8)) is significantly reduced due to the smaller matrix \( B_p \), compared to \( B \). The constrained optimization problem in (9) now changes to a set of problems, which are solved subsequently for all \( p \in \bar{p} \):

Algorithm 1 assigning carriers to subsets

1: \( J_1 = \{1\}; p = 1; \)
2: for \( j = 2, 3, \ldots, n \) do
3: if \( c_{(j-1)k} = 1 \) and \( c_{jk} = 1 \) for any \( k \in \bar{m} \) then
4: \( J_p = J_p \cup \{j\}; \)
5: else
6: \( p = p + 1; J_p = \{j\}; \)
7: end if
8: end for
9: \( P = p; \)

Algorithm 2 assigning actuators to subsets

1: for \( p \in \{1, 2, \ldots, P\} \) do
2: \( K_p = \emptyset; \)
3: for \( k \in \bar{m} \) do
4: if \( b_a(z_{aj}) \neq 0 \) or \( b_b(z_{bj}) \neq 0 \) for any \( j \in J_p \) then
5: \( K_p = K_p \cup \{k\}; \)
6: end if
7: end for
8: end for
IV. EXPERIMENTAL CASE STUDY

In this section, we present an experimental study on the industrial IPMLSM-based transportation system presented in Fig. 1. In particular, we show the performance improvements of the proposed control allocation strategy, compared to the state-of-practice control solution currently applied by the manufacturer. In this experimental study, we show the unconstrained, distributed allocation procedure as discussed in Section III-C, to illustrate the achievable performance improvement. The constrained allocation procedure has been illustrated by a simulation study on the same system in [13].

The considered system consists of two carriers and six actuators (cf. Fig. 1), and is modeled by (1)-(3) with \( n = 2 \) and \( m = 6 \). The carrier mass is 10.8 kg. The system is driven by an industrial precision motion control system (NYCe 4000), developed by the same manufacturer. All experiments are carried out at a sampling rate of 4kHz, and, using the decentralized approach in Section III-C, the allocation problem can be solved well within each sampling interval. The actuators are spaced such that the center distance between subsequent actuators is equal to the carrier length (cf. Fig. 3), unless specifically stated otherwise. Both the state-of-practice control strategy and the allocation strategy use a dedicated high-level (loop-shaped) motion controller for each carrier separately. We use the same motion controller (identical to the one designed by the manufacturer) for both carriers, consisting of a feedback term (lead filter, integrator, and a lowpass filter), and a feedforward term. The feedback term for each carrier in the Laplace domain is given by

\[
\tau_{fb,j}(s) = K \frac{1500s^2+1.1\times10^5s+1.58\times10^6}{1.1\times10^{-4}s^4+0.23s^3+236.7s^2+6.28\times10^4s} e_j(s),
\]

with \( s \in \mathbb{C} \), \( e_j = r_j - y_j \) the position error signal of carrier \( j \), and \( r_j \) the position reference of carrier \( j \). The stabilizing feedback controller achieves the specified crossover frequency of 20 Hz and satisfies standard robustness margins. The feedforward term consists of an acceleration feedforward term, a Coulomb friction compensation term, and an experimentally obtained term \( F_{fb,j} \) that compensates for magnet interaction forces between carriers, i.e.,

\[
\tau_{ff,j} = 0.9M_j\dot{r}_j + K_p\text{sign}(\dot{r}_j) + F_{fb,j}.
\]

Although the Coulomb friction in the system slightly varies over time and may be position-dependent, the experimentally validated value \( K_p = 4.9 \) N is chosen such that no overcompensation of Coulomb friction takes place, but still compensates for a significant part of it. The total desired control force on carrier \( j \) is then given by \( \tau_{c,j} = \tau_{ff,j} + \tau_{fb,j} \). Both the state-of-practice controller as the proposed allocation scheme aim at implementing the desired control force \( \tau_{c,j} \) for each carrier in a different manner (leading to a different performance) as we will illustrate below.

A. State-of-practice control strategy

The state-of-practice control strategy currently used by the manufacturer operates in the \( dq0 \) reference frame (see Fig. 2), so that \( i_q \) is the only control input for each carrier (for the considered control system, the direct current \( i_d \) is controlled to zero). The force-generating component \( i_q \) is then obtained by dividing the desired control force \( \tau_c \) by a fixed motor gain \( B_q = 23.565 \) N/A, see Fig. 6. By employing a strict actuator spacing, actuator end-effects are balanced so that the sum of the motor gains of the active actuators is approximately constant, regardless of the carrier position on the tracks. This control signal is then applied to each actuator that commutes with the considered carrier by means of a selector. However, when an actuator influences two carriers, the selector implements the control signal for the carrier that overlaps the actuator the most.

If an actuator commutes with only a single carrier, then this control strategy works well, as the actuator indeed applies the desired control force on the carrier by applying the correct commutation angle (which is a function of the carrier position). This control strategy, however, does no longer work properly when an actuator influences two carriers. The correct commutation angle is then only applied for the most overlapping carrier, which results in a wrong commutation angle for the second carrier. As a result, there is a mismatch between the desired and implemented control force on the carrier that overlaps the actuator the least, compromising tracking performance. The implications of this fact are illustrated by the experimental results in Section IV-C below.

Remark 3. Although the state-of-practice control strategy may be improved by manual compensation of the mentioned disturbance forces for simple system layouts, the proposed allocation technique offers a systematic and scalable approach for robustly improving tracking performance also suitable for large-scale systems with many carriers and actuators.

B. Proposed allocation strategy

In contrast to the state-of-practice control strategy, the allocation scheme is applied in the \( \alpha,\beta \) reference frame. The coefficients of the position-dependent motor gain matrix \( B(j) \) in (3), (6) are obtained from FEM-based electromagnetic simulations, see [1], by measuring the relative position \( z_{jk} \) of carrier \( j \) with respect to actuator \( k \). We assume that all actuators are identical, see Assumption 2. The motor gains, as a function of \( z_{jk} \), are presented in Fig. 7, where the deterioration of the gains at both ends of the region of influence can be observed (i.e., the end-effects). The weighting matrices are \( Q = 10^4I \) and \( W = I \), with \( I \) the identity matrix of appropriate dimensions.

C. Comparative performance study

We have implemented both the state-of-practice and allocation control strategies on the experimental setup of Fig. 1. The
following scenarios are studied: a) tracking of a demanding motion profile for a single carrier, with strict actuator spacing; b) tracking of a motion profile for a single carrier, with relaxed actuator spacing; c) a parallel motion of two carriers; d) a complex motion of two carriers, combining independent and adjacent motion. The results for the scenarios b)-d) are presented in Fig. 8-10 below. In these figures, four subplots are presented which show the following: the top subplot shows the reference trajectories of the carriers, and the second subplot shows the active actuators at each time instant. In here, the time intervals where an actuator influences two carriers are indicated by gray patches (in Fig. 9 and 10). The third subplot shows the Euclidian norm of the position error $\|e\|_2$ (with $e = [e_1, e_2]^T$), and the fourth subplot shows a metric for the power consumption, given by $u^T W u$.

For scenario a), both control strategies perform well during the constant velocity phase, despite the fact that multiple actuators influence the carrier. This is due to strict actuator spacing, that results in a smooth transition of a carrier between actuators (see Section II-A), and the fact that there is only one carrier on the tracks in this scenario.

Scenario b) Consider Fig. 8, which shows the experimental results of a single carrier performing a back and forth motion, where the actuator spacing has been relaxed. In particular, the second actuator has been shifted to the right by 6 mm. A strict actuator spacing realizes that the sum of the motor gains of the active actuators is constant, regardless of the carrier position on the tracks. In the current scenario, this property is violated by the relaxed actuator spacing, and, as a result, the state-of-practice control strategy implements a force that deviates from the desired control force coming from the motion controller. As can be observed in the third subplot of Fig. 8, this results in increased position errors in the regions where multiple actuators are active. The proposed allocation scheme is instead able to adapt and realizes a low tracking error while minimizing power consumption. The increased position error at the start of the experiment and velocity reversals is due to any remaining Coulomb friction.

Scenario c) For this experiment, the strict actuator spacing has been restored. Consider Fig. 9. Two carriers perform an adjacent back and forth motion, with two time intervals at which actuator 2 influences both carriers. In these intervals, the state-of-practice controller results in a large position error, which is caused by the fact that the shared actuators are only able to take the correct commutation angle for the most overlapping carrier. The other carrier then experiences a control force that deviates from the desired control force coming from the motion controller, due to a mismatch in the correct and attained commutation angle. The proposed allocation scheme is instead able to adapt and find a control input such that the trajectory can be followed well with significantly reduced position errors. Moreover, as the bottom subplot indicates, the power usage is decreased as well.

Scenario d) Consider Fig. 10. In this experiment, two carriers perform a complex motion. The carriers move independently as well as adjacent, where actuator 2 influences both carriers during the majority of the experiment. As discussed in Scenario c), the actuator can only take the correct commutation angle for the most overlapping carrier when using the state-of-practice control strategy, which results in pronounced large position errors. The proposed allocation scheme is instead capable of adapting to such a situation by altering the current inputs, resulting in low position errors. This shows the capability of the system to perform independent motion tasks with high accuracy. During the adjacent motion (in the time span 2.7-5.8 s), the system behaves similarly to Scenario c),
where the allocation scheme indeed outperforms the state-of-practice control solution in terms of position accuracy, see the inset in the third subplot. Note that the left and right error peaks in the inset are due to friction.

D. Discussion

The experimental study shows that the proposed control allocation scheme (compared to the state-of-practice controller used by the manufacturer) achieves improved tracking performance, evident from the Root-Mean-Square (RMS) values of the norm of the error $\|e\|_2$ for the scenarios a)-d), presented in the following table:

<table>
<thead>
<tr>
<th>scenario</th>
<th>state-of-pr. alloc.</th>
<th>state-of-pr. alloc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>0.0507</td>
<td>0.0111</td>
</tr>
<tr>
<td>b)</td>
<td>0.0230</td>
<td>0.0097</td>
</tr>
<tr>
<td>c)</td>
<td>0.0185</td>
<td>0.0185</td>
</tr>
<tr>
<td>d)</td>
<td>0.2193</td>
<td>0.0291</td>
</tr>
</tbody>
</table>

Furthermore, the allocation scheme provides the possibility to take actuator limits into account. This is illustrated by a simulation study of Scenario d), where $i_{\alpha}$ and $i_\beta$ are limited to a challenging value of 0.4A. An extensive simulation study can be found in [13], which we briefly discuss here for the sake of completeness. The results in Fig. 11 illustrate that employing the constrained control allocation scheme results in decreased tracking performance, compared to the unconstrained case, but significant less actuator duty. On the other hand, the state-of-practice controller (without actuator constraints) results in very large position errors.

V. Conclusions

We have presented a control allocation framework for an industrial high-accuracy transportation and positioning system, which results in enhanced tracking, allows for independent motion of multiple carriers, is able to take actuator limits into account, and results in relaxed hardware design specifications. A decentralized allocation procedure is presented that enhances computational efficiency, such that the control allocation scheme is suitable for online implementation at high sampling rates, and facilitates scalability to larger system configurations. We have illustrated the benefits of the proposed control allocation strategy by means of an experimental study.

REFERENCES