Real-time Kinematic Modeling and Prediction of Human Joint Motion in a Networked Rehabilitation System

Wenlong Zhang, Xu Chen, Joonbum Bae, and Masayoshi Tomizuka

***Abstract*— In this paper, a networked-based rehabilitation system is introduced for lower-extremity tele-rehabilitation. In order to enable high-level motion planning of the rehabilitation robot in real-time for enhanced safety and appropriate human-robot interactions, a time series model is proposed to capture the kinematics of knee joint rotations. A major challenge in such a system is that measurement data might be delayed or lost due to wireless communication. With a delay and loss compensation mechanism, a modified recursive least square (mRLS) algorithm is applied for real-time modeling and prediction of knee joint rotations in the sagittal plane, and convergence of the proposed algorithm is studied. Simulation and experimental results are presented to verify the performance of the proposed algorithm.**

I. INTRODUCTION

In view of the rapidly increasing number of patients and elderly people who need physical therapy, there is a large demand for gait rehabilitation and assistive devices. Many companies and researchers have developed various kinds of assistive robots to facilitate the rehabilitation treatment and patients' daily life [1]–[3]. As rehabilitation systems become more sophisticated, mobility becomes an important issue, which motivates us to integrate network media into a rehabilitation system, as is shown in this paper.

In order to achieve smart rehabilitation treatment, it is important to understand the user's joint movement. This has motivated intensive studies of human motion capture and analysis with optical sensors [4], inertial sensors [5], and electromyography (EMG) sensors [6]. For lower-extremity motion analysis, joint kinematic model is often employed to detect gait phases [7] for identifying gait abnormality. It is also frequently used to estimate human motion intention [8] for trajectory planning of the assistive robot. Joint kinematics is also particularly useful in fall prediction to enhance the safety of a rehabilitation robot [9].

Motivated by the importance of joint kinematic modeling, an autoregressive integrated (ARI) model was built based on time series analysis to predict knee joint rotation in a network-based rehabilitation system [10]. While the proposed technique can provide a reliable human knee joint model for motion prediction, it is difficult to implement the algorithm online due to its computational complexity. It would be ideal if an online adaptive human motion model can be built based on new measurements from the user. Such a model is able to capture the change of walking dynamics more accurately and provide more insights to the user.

In this paper, an online adaptive knee joint rotation model is built based on a modified recursive least square (mRLS) algorithm. Since measurement data are transmitted over the wireless network, network-induced constraints, such as time delay and packet loss, need to be handled in the modeling process. A delay and loss compensator is thus proposed with the proof of convergence in this paper. Simulation and experimental results are demonstrated to verify the effectiveness of the proposed algorithm. The influences of time delay, packet loss, and forgetting factors in the algorithm are analyzed based on the simulation and experimental results.

The remainder of this paper is organized as follows. In Section II, the networked rehabilitation system is briefly introduced and joint angle measurement for one healthy subject is presented. Section III proposes the mRLS algorithm for joint rotation modeling and illustrates how it can handle the network-induced challenges. Convergence analysis is presented in Section IV. Section V shows simulation results and analyzes the performance of the proposed approach. Experimental results with another healthy subject are shown in Section VI. Conclusion and future work are given in Section VII.

II. JOINT ANGLE MEASUREMENT IN A REHABILITATION SYSTEM

In our previous work [11], a network-based rehabilitation system was proposed for improved mobility and in-home tele-rehabilitation. The system consists of a wireless body sensor network and a rehabilitation robot controlled over a high-speed wireless network. A computer at the patient's home wirelessly connects to all sensors and design the proper control signals for the robot. The proposed algorithm will be implemented in the local computer.

In the proposed system, human joint rotation in three dimensions can be captured by several inertial sensors. A wireless inertial measurement unit (IMU) is shown in Fig. 1. An IMU node consists of a three degrees of freedom accelerometer, magnetometer, and gyroscope. The measurement data can be transmitted to the local computer wirelessly. The IMU node is powered by a Li-Po battery and it can work continuously for 90 minutes. The dimension of one wireless IMU node is 2 inches $\times$ 1.4 inches $\times$ 0.6 inches and its weight is around 0.15 lbs including the battery. The wireless

This work was supported by National Science Foundation under Grant CMMI-1013657.

W. Zhang and M. Tomizuka are with the Department of Mechanical Engineering, University of California, Berkeley, CA 94720 USA (e-mail: $wlzhang@berkeley.edu$; $tomizuka@me.berkeley.edu$)

X. Chen is with the Department of Mechanical Engineering, University of Connecticut, Storrs, CT 06269 USA (e-mail: $xchen@engr.uconn.edu$)

J. Bae is with the School of Mechanical and Nuclear Engineering, Ulsan National Institute of Science and Technology, Ulsan 689-798, Korea (e-mail: $jbbae@unist.ac.kr$)

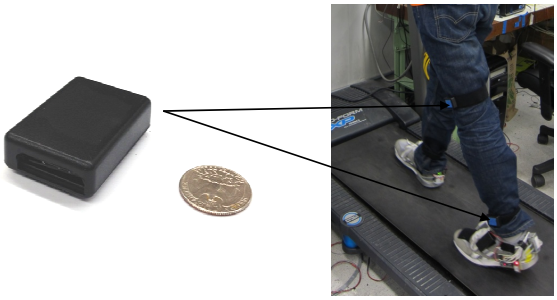


Fig. 1: An inertial measurement unit (IMU) and a human subject walking on a treadmill with IMUs

IMU can achieve a sampling rate up to 100Hz and it is very convenient to be attached to a subject's lower extremities using velcros, as is shown in Fig. 1.

To examine the performance of the proposed algorithm, several sets of knee joint rotation measurements were collected. With two IMUs attached to his left thigh and shank, a 25-year old male user was asked to walk on a treadmill at 1 mph in the Mechanical Systems Control Laboratory at the University of California, Berkeley. The user had no known walking abnormalities. The measurement data were recorded at 50 Hz and representative knee joint angles in the sagittal plane is shown in Fig. 2. The measurement data will be used in the modeling and simulation study.

III. HUMAN MOTION MODELING WITH A MODIFIED RECURSIVE LEAST SQUARE METHOD

A. Problem Formulation and Algorithm Design

In this paper, the following time-varying linear kinematic model is built

$$y(k) = \Phi^T(k)\,\theta(k) + v(k), \tag{1}$$

where $y(k)$ is the measured human joint angle and $\Phi(k) = \begin{bmatrix} y(k-1) & y(k-2) & \cdots & y(k-n) \end{bmatrix}^T \in \mathbb{R}^n$ is the regressor vector that stores the previous measurement of the joint angles, $\theta(k) = \begin{bmatrix} \theta_1(k) & \theta_2(k) & \cdots & \theta_n(k) \end{bmatrix}^T \in \mathbb{R}^n$ is a time-varying parameter vector that needs to be estimated. $v(k)$ is white noise with zero mean and variance σ_v^2, and it is independent of the current and previous regressors, i.e.,

$$E[v(k)] = 0, E\left[v^2(k)\right] = \sigma_v^2 < \infty, \tag{2}$$
$$E[v(k)\,v(j)] = 0, \forall k \neq j, \tag{3}$$
$$E[\Phi(k-i)\,v(k)] = \mathbf{0}, \forall i \geq 0. \tag{4}$$

The proposed model (1) is an autoregressive model, and its capability of capturing human joint kinematics was validated in [10]. In order to estimate the parameter vector, the following least square cost function is considered:

$$J(k) = \min_{\hat{\theta}(k)} \left\{ \frac{1}{2} \sum_{j=1}^{k} \lambda^{k-j} \left[y(j) - \Phi^T(j)\,\hat{\theta}(k) \right]^2 \right\}. \tag{5}$$

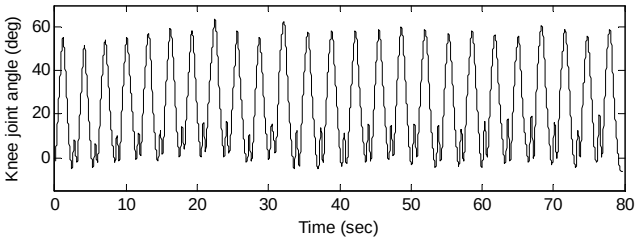


Fig. 2: Knee joint angle measurement

This problem can be solved recursively as follows:

$$e^o(k) = y(k) - \Phi^T(k)\,\hat{\theta}(k), \tag{6}$$
$$F(k) = \frac{1}{\lambda}\left[F(k-1) - \frac{F(k-1)\Phi(k)\Phi^T(k)F(k-1)}{\lambda + \Phi^T(k)F(k-1)\Phi(k)} \right], \tag{7}$$
$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\,\Phi(k)\,e^o(k), \tag{8}$$

where $e^o(k)$ is the a-priori estimation error and $F(k)$ is the adaptation gain matrix. Note that (7) is equivalent to $F^{-1}(k) = \lambda F^{-1}(k-1) + \Phi(k)\,\Phi^T(k)$. λ is a constant forgetting factor that needs to be selected prior to the modeling process. Furthermore, it is assumed that the true parameters follow a random walk process as follows:

$$\theta(k+1) = \theta(k) + w(k), \tag{9}$$

where $w(k)$ is a white noise with zero mean and variance σ_w^2, and it is independent of the true parameters, current and previous regressors, and process noise, i.e.,

$$E[w(k)] = \mathbf{0}, E\left[w^T(k)\,w(k)\right] = \sigma_w^2 < \infty, \tag{10}$$
$$E\left[w(k)\,w^T(j)\right] = \mathbf{0}, \forall k \neq j, \tag{11}$$
$$E\left[\Phi(k-i)\,w^T(k)\right] = \mathbf{0}, \forall i \geq 0, \tag{12}$$
$$E[w(k)\,v(i)] = \mathbf{0}, \forall i. \tag{13}$$

The random walk assumption of the true model parameters follows the observation that the gait pattern of a subject is consistent between steps with the same walking speed and road condition. However, there must be some minor differences of walking behaviors between steps. The differences are small for healthy subjects but large for patients with gait abnormalities. Therefore, σ_w^2 is typically small for healthy subjects and large for patients.

B. Model Structure Selection

In this paper, the model to be built has the form (1), and the order of the model is firstly determined. It is shown in [10] that one can decide the order of the model using Box-Jenkins approach [12]. Applying this approach to the data shown in Fig. 2 suggests a ninth-order autoregressive model with a first-order integration action. Thus, the order of the proposed model is picked as 10, i.e., $\Phi(k) \in \mathbb{R}^{10}$ and $\theta(k) \in \mathbb{R}^{10}$.

C. Time Delay and Packet Loss

In a network-based rehabilitation system, sensing packet is transmitted to the estimator over the wireless network. Time delay may happen during data transmission, making the most recent sensing packet unavailable for parameter update. A

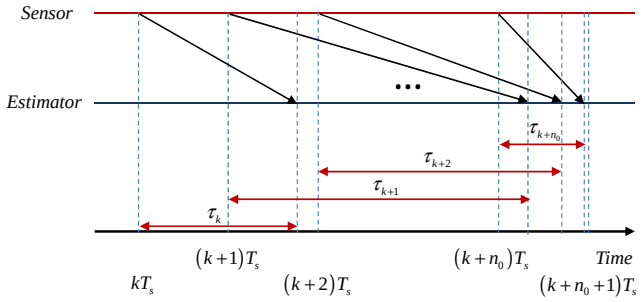


Fig. 3: Timing diagram of the measurement packet

representative timing diagram of the sensing packet delivery is shown in Fig. 3, where τ_k is the delay of the k^{th} measurement packet and T_s is the sampling interval of the system. Similarly, packet loss of sensing packets can also happen randomly, which leads to irregular parameter update and makes the convergence analysis challenging.

Time delay and packet loss have two major negative effects to the parameter estimation. On one hand, the measurement $y(k)$ might be unavailable for calculating $e^o(k)$ using (6). On the other hand, delayed or lost packet might need to be used in the regressor $\Phi(k)$ to update model parameters and the adaption gain.

In order to deal with varying time delay during network transmission, the following compensation mechanism is used. If an output measurement $y(k)$ is not available, the predicted output $\hat{y}(k)$ from the identified model is used, and the parameter will not be updated for this step. The identified model will also be used to predict missing elements in the regressor $\Phi(k)$. The mRLS algorithm can be expressed as

$$e^o(k) = z(k) - \Phi_e^T(k)\hat{\theta}(k), \tag{14}$$

$$F(k) = \frac{1}{\lambda}\left[F(k-1) - \frac{F(k-1)\Phi_e(k)\Phi_e^T(k)F(k-1)}{\lambda+\Phi_e^T(k)F(k-1)\Phi_e(k)}\right], \tag{15}$$

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\Phi_e(k)e^o(k), \tag{16}$$

where $z(k) = \gamma(k)y(k) + [1-\gamma(k)]\hat{y}(k)$, $\hat{y}(k) = \Phi_e^T(k)\hat{\theta}(k)$, and $\gamma(k) \sim Bernoulli(\gamma)$ is an indicator of successful transmission of the measurement packet $y(k)$.

$$\Phi_e(k) = \begin{bmatrix} s(k-1) & s(k-2) & \cdots & s(k-n) \end{bmatrix}^T \in \mathbb{R}^n,$$

where

$$s(k-i) = \beta_{k-i}(k)y(k-i) + [1-\beta_{k-i}(k)]\hat{y}(k-i).$$

$\beta_{k-i}(k)$ is an availability indicator of measurement packet $y(k-i)$ at the k^{th} time step. Note that a packet will be available in the regressor several steps later if it is delayed, but it will not be available if it is lost without retransmission.

IV. CONVERGENCE ANALYSIS

In this section, performance of the proposed mRLS algorithm is examined analytically by deriving the upper bound of the parameter estimation errors. Based on system model (1), assumptions for $v(k)$ and $w(k)$ in (2)-(4) and (10)-(13), and mRLS algorithm (14)-(16), the following further assumptions are made for derivation of the error bounds [13]:

(A1) For some constants $0 < \alpha \leq \beta < \infty$ and an integer $N \geq n$, the following strong persistent excitation condition holds:

$$\alpha I_n \leq \frac{1}{N+1}\sum_{i=0}^{N}\Phi_e(k+i)\Phi_e^T(k+i) \leq \beta I_n, \tag{17}$$

where I_n is an n-dimensional identity matrix.

(A2) The estimator gain matrix is initialized as $F(-1) = p_0 I_n$, where $\frac{1-\lambda}{(N+1)\beta} \leq p_0 \leq \frac{1-\lambda}{\alpha}$.

(A3) The parameter estimate is initialized as $\hat{\theta}(0) = c_0\mathbf{1}_n$, where c_0 is a small positive number and $\mathbf{1}_n$ is an n-dimensional vector with all elements equal to 1. In addition, $\hat{\theta}(0)$ is independent of $v(k)$.

(A4) Parmeter estimation error is defined as

$$\tilde{\theta}(k) = \hat{\theta}(k) - \theta(k), \tag{18}$$

where $\tilde{\theta}(0)$ satisfies $\mathbb{E}\left[\tilde{\theta}^T(0)\tilde{\theta}(0)\right] = \delta_0 < \infty$.

The assumptions above lead to the following lemma:

Lemma 1: For the system described in (1) and mRLS algorithm expressed in (14)-(16), if assumption (A1) holds, then for $k \geq N$ and $0 < \lambda < 1$ the adaption gain matrix $F(k)$ satisfies

$$\frac{\lambda^N\alpha}{1-\lambda}I_n + \lambda^{k+1}\left[F^{-1}(-1) - \frac{\alpha}{1-\lambda}I_n\right] \leq F^{-1}(k) \leq \frac{(N+1)\beta}{1-\lambda}I_n + \lambda^{k+1}\left[F^{-1}(-1) - \frac{(N+1)\beta}{1-\lambda}I_n\right]. \tag{19}$$

Proof: The proof is similar to that of Lemma 1 in [13] and is therefore omitted. ■

Note from Lemma 1 that if both (A1) and (A2) are satisfied, the following inequality is satisfied

$$\frac{1-\lambda}{(N+1)\beta}I_n \leq F(k) \leq \frac{1-\lambda}{\lambda^N\alpha}I_n \tag{20}$$

for $k \geq N$. Now the theorem that addresses the upper bounds of parameter estimate errors can be provided.

Theorem 1: Consider the system described in (1), mRLS algorithm expressed in (14)-(16), and $v(k)$ and $w(k)$ satisfying (2)-(4) and (10)-(13), respectively. If the assumptions (A1)-(A4) are satisfied, the expected norm of estimation error $\tilde{\theta}(k)$ satisfies the following upper bound for all $N \leq k < \infty$.

$$\mathop{\mathbb{E}}_{\substack{\tilde{\theta}(0),\\ \gamma(1),\cdots,\gamma(k),\\ \beta_0(1),\cdots,\beta_k(k)}}\left[\left\|\tilde{\theta}(k)\right\|^2\right] \leq \alpha^{-2}p_0^2\lambda^{2(k-N+1)}(1-\lambda)^2\delta_0 + \frac{1-\lambda}{\lambda^N\alpha}nB(k) + \frac{n(1-\lambda)}{\alpha\lambda^{N-1}}\sigma_v^2 + \frac{(N+1)^2\beta^2}{\alpha^2\lambda^{2(N-1)}(1-\lambda)^2}\sigma_w^2. \tag{21}$$

where α and β are defined in (A1), and $B(k)$ depends on the performance of network transmission.

Proof: Based on definition in (18) and mRLS algorithm (14)-(16), the following equations are obtained:

$$\begin{aligned}\tilde{\theta}(k+1) &= \hat{\theta}(k+1) - \theta(k+1) \\ &= \hat{\theta}(k+1) - [\theta(k) + w(k)] \\ &= \tilde{\theta}(k) + F(k)\Phi_e(k)v(k) - F(k)\Phi_e(k)\Phi_e^T(k)\tilde{\theta}(k) \\ &\quad + F(k)\Phi_e(k)[\Phi(k) - \Phi_e(k)]^T\theta(k) - w(k).\end{aligned} \tag{22}$$

By defining $\Delta(k) = \Phi_e(k) - \Phi(k)$, the equation above can be written as

$$\begin{aligned}\tilde{\theta}(k+1) &= \tilde{\theta}(k) - w(k) + F(k)\Phi_e(k)v(k) \\ &\quad - F(k)\Phi_e(k)\Phi_e^T(k)\tilde{\theta}(k) - F(k)\Phi_e(k)\Delta^T(k)\theta(k) \\ &= \left[I_n - F(k)\Phi_e(k)\Phi_e^T(k)\right]\tilde{\theta}(k) + F(k)\Phi_e(k)v(k) \\ &\quad - F(k)\Phi_e(k)\Delta^T(k)\theta(k) - w(k).\end{aligned} \tag{23}$$

Note (15) yields $\Phi_e(k)\Phi_e^T(k) = F^{-1}(k) - \lambda F^{-1}(k-1)$. Thus, the equation above can be rewritten as

$$\begin{aligned}\tilde{\theta}(k+1) &= \lambda F(k)F^{-1}(k-1)\tilde{\theta}(k) + F(k)\Phi_e(k)v(k) \\ &\quad -F(k)\Phi_e(k)\Delta^T(k)\theta(k) - w(k).\end{aligned} \tag{24}$$

Induction of (15) and (24) yields

$$F^{-1}(k-1) = \lambda F^{-1}(k-2) + \Phi_e(k-1)\Phi_e^T(k-1), \tag{25}$$

$$\begin{aligned}\tilde{\theta}(k) &= \lambda F(k-1)F^{-1}(k-2)\tilde{\theta}(k-1) \\ &\quad + F(k-1)\Phi_e(k-1)v(k-1) - w(k-1) \\ &\quad - F(k-1)\Phi_e(k-1)\Delta^T(k-1)\theta(k-1).\end{aligned} \tag{26}$$

Plugging in $F^{-1}(k-1)$ and $\tilde{\theta}(k)$ into (24) results in the following equation

$$\begin{aligned}\tilde{\theta}(k+1) &= \lambda^2 F(k)F^{-1}(k-2)\tilde{\theta}(k-1) \\ &\quad - F(k)\Phi_e(k)\Delta^T(k)\theta(k) \\ &\quad - \lambda F(k)\Phi_e(k-1)\Delta^T(k-1)\theta(k-1) \\ &\quad + \lambda F(k)\Phi_e(k-1)v(k-1) + F(k)\Phi_e(k)v(k) \\ &\quad - \lambda F(k)F^{-1}(k-1)w(k-1) - w(k).\end{aligned} \tag{27}$$

Continuing the induction yields

$$\begin{aligned}\tilde{\theta}(k+1) &= \lambda^{k+1}F(k)F^{-1}(-1)\tilde{\theta}(0) \\ &\quad - \sum_{i=0}^{k}\lambda^{k-i}F(k)\Phi_e(i)\Delta^T(i)\theta(i) \\ &\quad - \sum_{i=0}^{k}\lambda^{k-i}F(k)F^{-1}(i)w(i) \\ &\quad + \sum_{i=0}^{k}\lambda^{k-i}F(k)\Phi_e(k)v(i) \\ &= \alpha_1(k) + \alpha_2(k) + \alpha_3(k) + \alpha_4(k),\end{aligned} \tag{28}$$

where the upper bounds of expected 2-norm of $\alpha_1(k)$, $\alpha_3(k)$, and $\alpha_4(k)$ can be similarily calculated as shown in [13]. For the bounds of $\alpha_2(k)$, the following transformation is performed:

$$\begin{aligned}\alpha_2(k) &= -\sum_{i=0}^{k}\lambda^{k-i}F(k)\Phi_e(i)\Delta^T(i)\theta(i) \\ &= -F(k)G^T(k)D(k),\end{aligned} \tag{29}$$

where

$$G(k) = \begin{bmatrix}\Phi_e^T(k) \\ \mu\Phi_e^T(k-1) \\ \vdots \\ \mu^k\Phi_e^T(0)\end{bmatrix} \in \mathbb{R}^{(k+1)\times n}, \mu := \sqrt{\lambda},$$

$$D(k) = \begin{bmatrix}\Delta^T(k)\theta(k) \\ \mu\Delta^T(k-1)\theta(k-1) \\ \vdots \\ \mu^k\Delta^T(0)\theta(0)\end{bmatrix} \in \mathbb{R}^{k+1}.$$

The following assumption is made

$$\mathop{\mathbb{E}}_{\beta_{k-1}(k),\cdots,\beta_{k-n}(k)}\left[D(k)D^T(k)\right] \le B(k)$$

since both $\Delta(k)$ and $\theta(k)$ are finite as long as $k < \infty$. Note that $F^{-1}(k) = G^T(k)G(k) + \lambda^{k+1}F^{-1}(-1)$. This equation yields

$$\begin{aligned}tr\left[F(k)G^T(k)G(k)\right] &= tr\left[I_n - \lambda^{k+1}F(k)F^{-1}(-1)\right] \\ &\ge tr\left[\left(1 - \frac{(1-\lambda)\lambda^{k+1}}{\lambda^N\alpha p_0}\right)I_n\right] \\ &= n\left[1 - \alpha^{-1}p_0^{-1}\lambda^{k-N+1}(1-\lambda)\right]\end{aligned} \tag{30}$$

based on (20). Similarly, one can get

$$\begin{aligned}tr\left[F(k)G^T(k)G(k)\right] &= tr\left[I_n - \lambda^{k+1}F(k)F^{-1}(-1)\right] \\ &= tr\left[I_n - \frac{\lambda^{k+1}F(k)}{p_0}\right] \le n.\end{aligned} \tag{31}$$

In this case, the following upper bound of $\mathbb{E}\left[\left\|\alpha_2(k)\right\|^2\right]$ can be derived (the subscript of expectation is omitted)

$$\begin{aligned}\mathbb{E}\left[\left\|\alpha_2(k)\right\|^2\right] &= \mathbb{E}\left\{tr\left[F(k)G^T(k)D(k)D^T(k)G(k)F(k)\right]\right\} \\ &\le tr\left\{\mathbb{E}\left[F(k)G^T(k)G(k)\right]\right\}\frac{1-\lambda}{\lambda^N\alpha}B(k) \\ &\le \frac{1-\lambda}{\lambda^N\alpha}nB(k).\end{aligned} \tag{32}$$

Therefore the estimation error is bounded. Proof is complete. ■

It is clear that the performance of parameter inference depends on the bound $B(k)$, which is a function of the time delay and packet loss. In the next section, simulations will be conducted to examine the influence of networked-induced constraints to parameter adaption and motion prediction.

V. SIMULATION STUDY

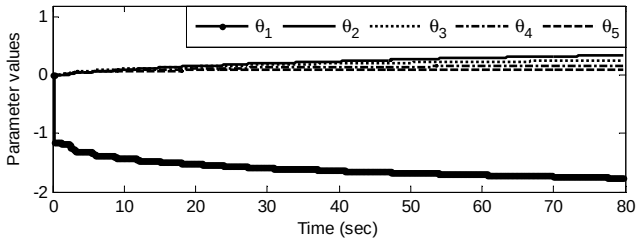
A. Simulation Results without Network-induced Constraints

In this subsection, the proposed mRLS algorithm is implemented without any network-induced constraints. The knee joint rotation data shown in Fig. 2 were used to build a 10^{th} order linear model and the parameter adaption result with a forgetting factor $\lambda = 1$ is shown in Fig. 4, from which one can observe that the coefficient for the most recent measurement (θ_1) dominates. Moreover, the parameters did not converge even at the end of the simulation. In order to achieve a faster convergence of the model parameters, forgetting factors are introduced into parameter adaption. Parameter adaption result with a forgetting factor $\lambda = 0.997$ is shown in Fig. 5, which confirms the convergence of the model parameters. Comparing with Fig. 4, one can conclude that introducing forgetting factors into the mRLS algorithm leads to faster parameter convergence with fluctuations.

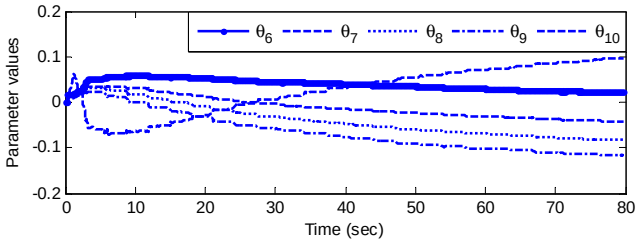
Using the identified model shown in Fig. 5, a 5-step prediction is performed to examine the accuracy of the model and prediction errors are shown in Fig. 6. It is clear that the prediction error is less than 1 degree, which verifies the performance of the identified model. As a baseline algorithm, the simplest way of 5-step prediction is to use the current measurement as a rough estimate, which is equivalent to

Max. Delay	γ	λ				
		0.993	0.995	0.997	0.999	1
0	100%	0.1875	0.1891	0.1918	0.1985	0.2078
$2T_s$	95%	0.6422	0.6368	0.6369	0.6475	0.6732
$2T_s$	90%	1.0002	0.9088	0.8653	0.8165	0.8017
$4T_s$	90%	2.1186	2.1713	2.0135	1.8206	1.7627

TABLE I: Root-mean-square (RMS) errors of 5-step prediction with different forgetting factors (deg)



(a) Adaption of the first five model parameters



(b) Adaption of the last five model parameters

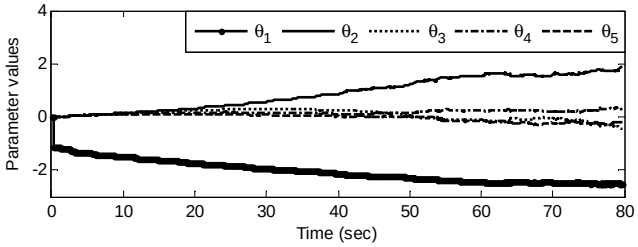
Fig. 4: Adaption of model parameters without forgetting factors

a zero-order hold (ZOH) and it yields a root-mean-square (RMS) error of 5.0146 degrees. In order to further examine the effectiveness of the forgetting factors, the 5-step prediction errors of the identified model with different forgetting factors are shown in the first row of Table I, which verifies the performance improvement brought by the forgetting factors. It is also verified that the prediction errors become smaller as forgetting factors decrease.

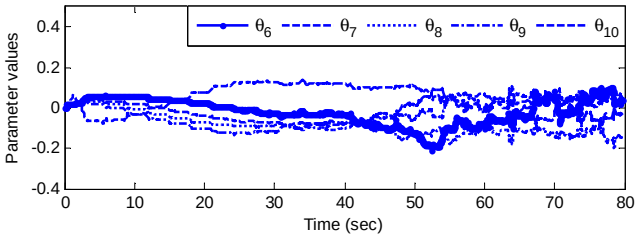
It is noticed that the forgetting factors are chosen to be very close to 1 in the simulations, which is due to the nature of the human walking pattern. Since walking patterns in each gait cycle cannot be the same, the true kinematic model is also varying. Moreover, there are inevitably unexpected walking patterns and measurement noises in the parameter adaption process. Thus, although choosing small forgetting factors might make the parameters converge faster, it will make the algorithm aggressive and result in large oscillations in the parameter adaption. Applying a low-pass filter to the measurement data might reduce such oscillations, but it might eliminate useful information in the measurements.

B. Simulation Results with Time Delay and Packet Loss

This section shows the simulation results of model identification with time delay and packet loss. We chose the successful transmission rate γ to be 90% and 95%, respectively, and the time delay of the k^{th} packet $\tau_k \sim unif\,(0, 4T_s)$ and $\tau_k \sim unif\,(0, 2T_s)$ respectively. Parameter adaption result of



(a) Adaption of the first five model parameters



(b) Adaption of the last five model parameters

Fig. 5: Adaption of model parameters with a forgetting factor $\lambda = 0.997$

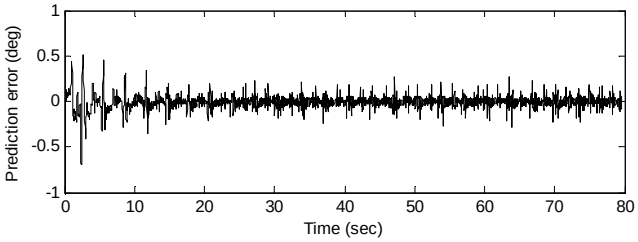


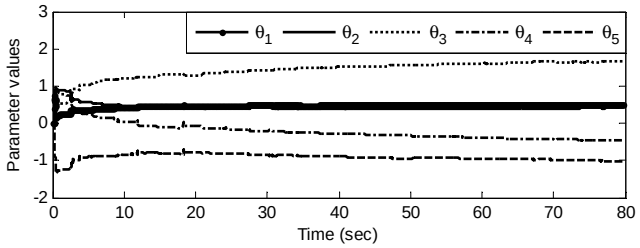
Fig. 6: Performance of 5-step prediction using identified linear model ($\lambda = 0.997$)

the proposed algorithm with $\gamma = 95\%$, $\tau_k \sim unif\,(0, 2T_s)$, and $\lambda = 0.997$ is shown in Fig. 7. It is verified that the parameters converge even with packet loss and time delay, but the identified model parameters are quite different.

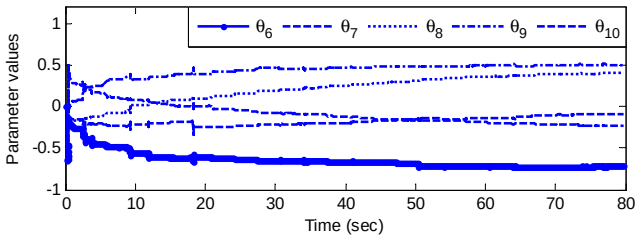
The effectiveness of forgetting factors is a bit more complicated in the cases with network-induced constraints, as is shown in Table I. When the time delay and packet loss are not very significant (the case with a maximum of two-step delay and $\gamma = 95\%$), reducing forgetting factors may lead to an improvement of the prediction accuracy. However in this case, when the forgetting factor is reduced to 0.993, the prediction error is even larger than the case with a forgetting factor of 0.995. Moreover, when the packet loss and time delay become more severe, reducing forgetting factors will lead to large fluctuations of the adapted parameters and prediction errors may be even larger than the case without forgetting factors. To summarize, larger forgetting factors need to be chosen with network-induced constraints.

VI. EXPERIMENTAL RESULTS

In this section, the proposed system is implemented in LabVIEW to build the knee joint kinematic model in real-time. The wireless IMU sensors were used to measure knee joint rotation angles in the sagittal plane. The sampling rate

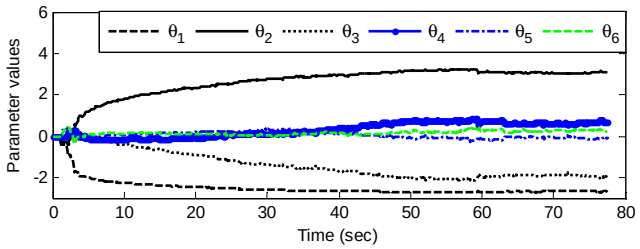


(a) Adaption of the first five model parameters

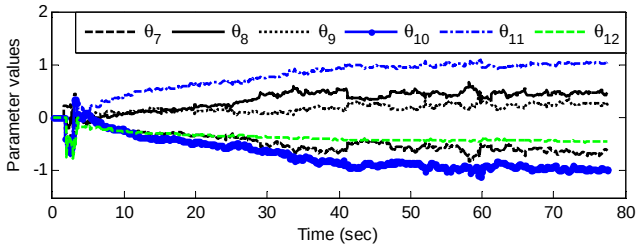


(b) Adaption of the last five model parameters

Fig. 7: Adaption of model parameters with a maximum two-step delay, γ=95%, and $\lambda = 0.997$



(a) Adaption of the first six model parameters



(b) Adaption of the last six model parameters

Fig. 8: Adaption of model parameters in the experiment with a forgetting factor $\lambda = 0.995$

was chosen to be 50Hz and there might be time delay and packet loss due to the wireless transmission of the sensor signals. In order to verify the performance of the proposed algorithm, a different subject was selected (a 21-year old male without known walking abnormalities) and he was asked to walk on a treadmill at 2 mph. A pre-trial was conducted for 10 seconds and the recorded data were used to determine the order of the model. Based on the techniques presented in Section III-B, the order was determined to be 12. In the formal experiment the subject was asked to walk on the treadmill with the same speed.

The identified model parameters for the first 80 seconds of formal experiment are shown in Fig. 8. The forgetting factor was chosen as 0.995. It is evident that all parameters finally converge. The identified model was used to achieve 5-step prediction, and the RMS prediction error of the proposed algorithm is 2.4027 degrees. The 5-step RMS prediction error from the baseline algorithm is 11.2674 degrees due to the increased speed, from which the effectiveness of the proposed algorithm is confirmed.

VII. CONCLUSION

In this paper, a networked rehabilitation system was introduced for lower-extremity rehabilitation. In order to enable high-level motion planning for enhanced safety and appropriate human-robot interactions, a time series model was proposed to describe the knee joint rotation. In view of time delay and packet loss of the measurement packets, a modified recursive least square (mRLS) algorithm was proposed for real-time modeling of the knee joint rotation and convergence of the proposed algorithm was studied. Effectiveness of the time delay, packet loss, and forgetting factors was verified in simulations and experiments.

The proposed algorithm has many potentially important applications, one of which is to achieve fall prediction based on a fast change of identified model parameters. Another application is to achieve high-level trajectory planning of the rehabilitation robot based on the predicted human motion. One of the ongoing work is to get the data from stroke and Parkinson's disease patients with walking abnormalities. Model orders and parameters of the patients will be compared and analyzed.

REFERENCES

[1] "Lokomat," `http://www.hocoma.com/`.
[2] R. J. Farris, H. A. Quintero, and M. Goldfarb, "Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 19, no. 6, pp. 652–659, 2011.
[3] S. K. Banala, S. Kim, S. K. Agrawal, and J. P. Scholz, "Robot assisted gait training with active leg exoskeleton (ALEX)," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 1, pp. 2–8, 2009.
[4] "Vicon Bonita," `http://www.vicon.com/`.
[5] "Xsens MVN," `http://www.xsens.com/`.
[6] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, and J. Yang, "A framework for hand gesture recognition based on accelerometer and emg sensors," *IEEE Trans. Syst., Man, Cybern. A*, vol. 41, no. 6, pp. 1064–1076, 2011.
[7] I. Pappas, T. Keller, S. Mangold, M. Popovic, V. Dietz, and M. Morari, "A reliable gyroscope-based gait-phase detection sensor embedded in a shoe insole," *IEEE Sensors J.*, vol. 4, no. 2, pp. 268–274, 2004.
[8] H. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 3, pp. 542–551, 2010.
[9] G. Wu and S. Xue, "Portable preimpact fall detector with inertial sensors," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 16, no. 2, pp. 178–183, 2008.
[10] W. Zhang, M. Tomizuka, and J. Bae, "Time series prediction of knee joint movement and its application to a network-based rehabilitation system," in *Proc. 2014 American Control Conf.*, 2014, pp. 4810–4815.
[11] W. Zhang, X. Zhu, S. Han, N. Byl, A. K. Mok, and M. Tomizuka, "Design of a network-based mobile gait rehabilitation system," in *Proc. Int'l Conf. on Robotics and Biomimetics*, 2012, pp. 1773–1778.
[12] G. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, Wiley. com, 2013.
[13] F. Ding and T. Chen, "Performance bounds of forgetting factor least-squares algorithms for time-varying systems with finite measurement data," *IEEE Trans. Circuits Syst. I*, vol. 52, no. 3, pp. 555–566, 2005.