

1 Introduction

The goal of this document is to derive an ILC update law that inherently considers the impact of the next iteration, $j + 1$ on the iteration after that, $j + 2$. The process will be similar to previous work in that we will start by forming a quadratic performance index, re-write that performance index in terms of only the design variable (sequence of control inputs), differentiate that performance index with respect to the design variable, set the result equal to the zero vector, then solve or rearrange to obtain an update law. This performance index will then include terms that depend on $u_j, u_{j+1}, u_{j+2}, x_j, x_{j+1}, x_{j+2}, e_j, e_{j+1}$ and e_{j+2} .

2 Preliminaries

In this work iteration j is the one which was just completed, iteration $j + 1$ is the iteration that is about to occur but has not yet begun. Also, assume the path is discretized into n_s steps, we have n_x state variables, and n_u control inputs.

Assume we are given the following a path-domain lifted system model of the form

$$\delta x_j = H_j \delta u_j + F_j \delta x_j^0 \quad (1)$$

where $\delta x_j \triangleq x_{j+1} - x_j$, $\delta u_j \triangleq u_{j+1} - u_j$ and $\delta x_j^0 \triangleq x_{j+1}^0 - x_j^0$. Here, x_j^0 and x_{j+1}^0 are the initial state vectors at iteration j and iteration $j + 1$ respectively. Therefore, another expression for the lifted system model is

$$x_{j+1} = x_j + H_j (u_{j+1} - u_j) + F_j (x_{j+1}^0 - x_j^0) \quad (2)$$

Note that for our application the final state vector from iteration j is the initial state vector for iteration $j + 1$. Therefore, we define E_F and E_I according to

$$E_F \triangleq \begin{bmatrix} \mathbb{0}^{n_x \times n_x(n_s-1)} & \mathbb{I}^{n_x \times n_x} \end{bmatrix} \quad (3)$$

$$E_I \triangleq \begin{bmatrix} \mathbb{I}^{n_x \times n_x} & \mathbb{0}^{n_x \times n_x(n_s-1)} \end{bmatrix} \quad (4)$$

so that

$$\hat{F}_j \triangleq F_j (E_F - E_I) \quad (5)$$

so that the expression for the lifted system model can be written as

$$x_{j+1} = x_j + H_j (u_{j+1} - u_j) + \hat{F}_j x_j \quad (6)$$

$$= x_j + \hat{F}_j x_j + H_j (u_{j+1} - u_j) \quad (7)$$

$$= (\mathbb{I} + \hat{F}_j) x_j + H_j (u_{j+1} - u_j) \quad (8)$$

$$= (\mathbb{I} + \hat{F}_j) x_j - H_j u_j + H_j u_{j+1} \quad (9)$$

$$= x_j + \hat{F}_j x_j - H_j u_j + H_j u_{j+1} \quad (10)$$

So this is our expression for x_{j+1} that will be used in the performance index later on. Next, lets find an expression for x_{j+2} . If we had already run iteration $j + 1$ then we could use a lifted system that looks like this

$$x_{j+2} = (\mathbb{I} + F_{j+1}) x_{j+1} + H_{j+1} (u_{j+2} - u_{j+1}) \quad (11)$$

however, we cant do that. Instead, we use the current model to approximate the behavior at iteration $j + 2$,

$$x_{j+2} = x_j + H_j (u_{j+2} - u_j) + F_j (x_{j+2}^0 - x_j^0) \quad (12)$$

where that x_{j+2}^0 is the final condition from iteration $j + 1$. Therefore, noting that

$$x_j^0 = E_I x_j \quad (13)$$

$$x_{j+2}^0 = E_F x_{j+1} \quad (14)$$

$$= E_F (x_j + \hat{F}_j x_j - H_j u_j + H_j u_{j+1}) \quad (15)$$

$$= E_F x_j + E_F \hat{F}_j x_j - E_F H_j u_j + E_F H_j u_{j+1} \quad (16)$$

makes the lifted model

$$x_{j+2} = x_j + H_j (u_{j+2} - u_j) + F_j \left(E_F x_j + E_F \hat{F}_j x_j - E_F H_j u_j + E_F H_j u_{j+1} - E_I x_j \right) \quad (17)$$

$$= x_j + H_j u_{j+2} - H_j u_j + F_j E_F x_j + F_j E_F \hat{F}_j x_j - F_j E_F H_j u_j + F_j E_F H_j u_{j+1} - F_j E_I x_j \quad (18)$$

$$= x_j + \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j - (F_j E_F + \mathbb{I}) H_j u_j + F_j E_F H_j u_{j+1} + H_j u_{j+2} \quad (19)$$

Next, lets look at the expressions for error sequences, starting with e_{j+1} ,

$$e_{j+1} = r - x_{j+1} \quad (20)$$

$$= r - (x_j + \hat{F}_j x_j - H_j u_j + H_j u_{j+1}) \quad (21)$$

$$= r - x_j - \hat{F}_j x_j + H_j u_j - H_j u_{j+1} \quad (22)$$

$$= e_j - \hat{F}_j x_j + H_j u_j - H_j u_{j+1} \quad (23)$$

Next, the expression for e_{j+2} .

$$e_{j+2} = r - x_{j+2} \quad (24)$$

$$= r - \left(x_j + \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j - (F_j E_F + \mathbb{I}) H_j u_j + F_j E_F H_j u_{j+1} + H_j u_{j+2} \right) \quad (25)$$

$$= r - x_j - \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j + (F_j E_F + \mathbb{I}) H_j u_j - F_j E_F H_j u_{j+1} - H_j u_{j+2} \quad (26)$$

$$= e_j - \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j + (F_j E_F + \mathbb{I}) H_j u_j - F_j E_F H_j u_{j+1} - H_j u_{j+2} \quad (27)$$

$$(28)$$

3 Writing the Performance Index

So, I'm going to shoot for a super-generic performance index that penalizes the values of and iteration-to-iteration deviations in the control input, state, and error. We also want to include a linear term on the states which can be used to incentive maximizing a given state. So then I'll choose:

$$J = u_{j+2}^T Q_u u_{j+2} + u_{j+1}^T Q_u u_{j+1} + (u_{j+2} - u_{j+1})^T Q_{\delta u} (u_{j+2} - u_{j+1}) + (u_{j+1} - u_j)^T Q_{\delta u} (u_{j+1} - u_j) \quad (29)$$

$$+ e_{j+2}^T Q_e e_{j+2} + e_{j+1}^T Q_e e_{j+1} + (e_{j+2} - e_{j+1})^T Q_{\delta e} (e_{j+2} - e_{j+1}) + (e_{j+1} - e_j)^T Q_{\delta e} (e_{j+1} - e_j) \quad (30)$$

$$+ x_{j+2}^T Q_x x_{j+2} + x_{j+1}^T Q_x x_{j+1} + (x_{j+2} - x_{j+1})^T Q_{\delta x} (x_{j+2} - x_{j+1}) + (x_{j+1} - x_j)^T Q_{\delta x} (x_{j+1} - x_j) \quad (31)$$

$$+ S_x x_{j+1} + S_x x_{j+2} \quad (32)$$

Start by multiplying everything out

$$J = u_{j+2}^T Q_u u_{j+2} + u_{j+1}^T Q_u u_{j+1} + u_{j+2}^T Q_{\delta u} u_{j+2} - 2u_{j+2}^T Q_{\delta u} u_{j+1} + u_{j+1}^T Q_{\delta u} u_{j+1} + u_{j+1}^T Q_{\delta u} u_{j+1} - 2u_{j+1}^T Q_{\delta u} u_j + u_j^T Q_{\delta u} u_j \quad (33)$$

$$+ e_{j+2}^T Q_e e_{j+2} + e_{j+1}^T Q_e e_{j+1} + e_{j+2}^T Q_{\delta e} e_{j+2} - 2e_{j+2}^T Q_{\delta e} e_{j+1} + e_{j+1}^T Q_{\delta e} e_{j+1} + e_{j+1}^T Q_{\delta e} e_{j+1} - 2e_{j+1}^T Q_{\delta e} e_j + e_j^T Q_{\delta e} e_j \quad (34)$$

$$+ x_{j+2}^T Q_x x_{j+2} + x_{j+1}^T Q_x x_{j+1} + x_{j+2}^T Q_{\delta x} x_{j+2} - 2x_{j+2}^T Q_{\delta x} x_{j+1} + x_{j+1}^T Q_{\delta x} x_{j+1} + x_{j+1}^T Q_{\delta x} x_{j+1} - 2x_{j+1}^T Q_{\delta x} x_j + x_j^T Q_{\delta x} x_j \quad (35)$$

$$+ S_x x_{j+1} + S_x x_{j+2} \quad (36)$$

Gather like terms

$$J = u_{j+2}^T (Q_u + Q_{\delta u}) u_{j+2} + u_{j+1}^T (Q_u + 2Q_{\delta u}) u_{j+1} - 2u_{j+2}^T Q_{\delta u} u_{j+1} - 2u_{j+1}^T Q_{\delta u} u_j + u_j^T Q_{\delta u} u_j \quad (37)$$

$$+ e_{j+2}^T (Q_e + Q_{\delta e}) e_{j+2} + e_{j+1}^T (Q_e + 2Q_{\delta e}) e_{j+1} - 2e_{j+2}^T Q_{\delta e} e_{j+1} - 2e_{j+1}^T Q_{\delta e} e_j + e_j^T Q_{\delta e} e_j \quad (38)$$

$$+ x_{j+2}^T (Q_x + Q_{\delta x}) x_{j+2} + x_{j+1}^T (Q_x + Q_{\delta x}) x_{j+1} - 2x_{j+2}^T Q_{\delta x} x_{j+1} - 2x_{j+1}^T Q_{\delta x} x_j + x_j^T Q_{\delta x} x_j \quad (39)$$

$$+ S_x x_{j+1} + S_x x_{j+2} \quad (40)$$

So now the goal is to write expressions for the state and error trajectories over the next *two* iterations in terms of the control input over the next two iterations. To do this, define some super-lifted vectors and some block matrices

$$\mathbf{u}_{k+1} \triangleq \begin{bmatrix} u_{j+1} \\ u_{j+2} \end{bmatrix} \quad \mathbf{x}_{k+1} \triangleq \begin{bmatrix} x_{j+1} \\ x_{j+2} \end{bmatrix} \quad \mathbf{e}_{k+1} \triangleq \begin{bmatrix} e_{j+1} \\ e_{j+2} \end{bmatrix} \quad \mathbf{r} \triangleq \begin{bmatrix} r \\ r \end{bmatrix} \quad (41)$$

$$\mathbf{Q}_u \triangleq \begin{bmatrix} Q_u + 2Q_{\delta u} & -Q_{\delta u} \\ -Q_{\delta u} & Q_u + Q_{\delta u} \end{bmatrix} \quad \mathbf{Q}_e \triangleq \begin{bmatrix} Q_e + 2Q_{\delta e} & -Q_{\delta e} \\ -Q_{\delta e} & Q_e + Q_{\delta e} \end{bmatrix} \quad \mathbf{Q}_x \triangleq \begin{bmatrix} Q_u + 2Q_{\delta x} & -Q_{\delta x} \\ -Q_{\delta x} & Q_u + Q_{\delta x} \end{bmatrix} \quad \mathbf{S}_x \triangleq \begin{bmatrix} S_x & S_x \end{bmatrix} \quad (42)$$

$$\mathbf{q}_{\delta u} \triangleq \begin{bmatrix} -2Q_{\delta u} & 0 \end{bmatrix} \quad \mathbf{q}_{\delta e} \triangleq \begin{bmatrix} -2Q_{\delta e} & 0 \end{bmatrix} \quad \mathbf{q}_{\delta x} \triangleq \begin{bmatrix} -2Q_{\delta x} & 0 \end{bmatrix} \quad (43)$$

so one iteration in the k -domain here would include two iterations in the j domain. Now write J in terms of these lifted-super vectors. We can now use these to re-write the performance index

$$J = \mathbf{u}_{k+1}^T \mathbf{Q}_u \mathbf{u}_{k+1} + u_j^T \mathbf{q}_{\delta u} \mathbf{u}_{k+1} + u_j^T Q_{\delta u} u_j + \mathbf{e}_{k+1}^T \mathbf{Q}_e \mathbf{e}_{k+1} + e_j^T \mathbf{q}_{\delta e} \mathbf{e}_{k+1} + e_j^T Q_{\delta e} e_j + \mathbf{x}_{k+1}^T \mathbf{Q}_x \mathbf{x}_{k+1} + x_j^T \mathbf{q}_{\delta x} \mathbf{x}_{k+1} + x_j^T Q_{\delta x} x_j + \mathbf{S}_x \mathbf{x}_{k+1} \quad (44)$$

So the goal will be to differentiate this with respect to \mathbf{u}_{k+1} but to do that, first we need to write \mathbf{x}_{k+1} and \mathbf{e}_{k+1} in terms of \mathbf{u}_{k+1}

$$\mathbf{x}_{k+1} = \begin{bmatrix} x_{j+1} \\ x_{j+2} \end{bmatrix} \quad (45)$$

$$= \begin{bmatrix} x_j + \hat{F}_j x_j - H_j u_j + H_j u_{j+1} \\ x_j + \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j - (F_j E_F + \mathbb{I}) H_j u_j + F_j E_F H_j u_{j+1} + H_j u_{j+2} \end{bmatrix} \quad (46)$$

$$= \begin{bmatrix} \mathbb{I} \\ \mathbb{I} \end{bmatrix} x_j + \begin{bmatrix} \hat{F}_j \\ F_j E_F + F_j E_F \hat{F}_j - F_j E_I \end{bmatrix} x_j - \begin{bmatrix} \mathbb{I} \\ F_j E_F + \mathbb{I} \end{bmatrix} H_j u_j + \begin{bmatrix} \mathbb{I} & 0 \\ F_j E_F & \mathbb{I} \end{bmatrix} H_j \mathbf{u}_{k+1} \quad (47)$$

$$= \mathbf{I} x_j + \mathbf{F}_j x_j + \mathbf{G}_j u_j + \mathbf{H}_j \mathbf{u}_{k+1} \quad (48)$$

$$\mathbf{e}_{k+1} = \begin{bmatrix} e_{j+1} \\ e_{j+2} \end{bmatrix} \quad (49)$$

$$= \begin{bmatrix} e_j - \hat{F}_j x_j + H_j u_j - H_j u_{j+1} \\ e_j - \left(F_j E_F + F_j E_F \hat{F}_j - F_j E_I \right) x_j + (F_j E_F + \mathbb{I}) H_j u_j - F_j E_F H_j u_{j+1} - H_j u_{j+2} \end{bmatrix} \quad (50)$$

$$= \begin{bmatrix} \mathbb{I} \\ \mathbb{I} \end{bmatrix} e_j - \begin{bmatrix} \hat{F}_j \\ F_j E_F + F_j E_F \hat{F}_j - F_j E_I \end{bmatrix} x_j + \begin{bmatrix} \mathbb{I} \\ F_j E_F + \mathbb{I} \end{bmatrix} H_j u_j - \begin{bmatrix} \mathbb{I} & 0 \\ F_j E_F & \mathbb{I} \end{bmatrix} H_j \mathbf{u}_{k+1} \quad (51)$$

$$= \mathbf{I} e_j - \mathbf{F}_j x_j - \mathbf{G}_j u_j - \mathbf{H}_j \mathbf{u}_{k+1} \quad (52)$$

with \mathbf{M}_j , \mathbf{F}_j , \mathbf{H}_j , and \mathbf{I} defined appropriately

These expressions mean that

$$\frac{d\mathbf{x}_{k+1}}{d\mathbf{u}_{k+1}} = \mathbf{H}_j \quad \text{and} \quad \frac{d\mathbf{e}_{k+1}}{d\mathbf{u}_{k+1}} = -\mathbf{H}_j \quad (53)$$

4 Deriving Learning Filters

So differentiating J gives:

$$\frac{dJ}{d\mathbf{u}_{k+1}} = 2\mathbf{u}_{k+1}^T \mathbf{Q}_u + u_j^T \mathbf{q}_{\delta u} - 2\mathbf{e}_{k+1}^T \mathbf{Q}_e \mathbf{H}_j - e_j^T \mathbf{q}_{\delta e} \mathbf{H}_j + 2\mathbf{x}_{k+1}^T \mathbf{Q}_x \mathbf{H}_j + x_j^T \mathbf{q}_{\delta x} \mathbf{H}_j + \mathbf{S}_x \mathbf{H}_j \quad (54)$$

Transposing everything

$$\frac{dJ}{d\mathbf{u}_{k+1}}^T = 2\mathbf{Q}_u \mathbf{u}_{k+1} + \mathbf{q}_{\delta u}^T u_j - 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{e}_{k+1} - \mathbf{H}_j^T \mathbf{q}_{\delta e}^T e_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{x}_{k+1} + \mathbf{H}_j^T \mathbf{q}_{\delta x}^T x_j + \mathbf{H}_j^T \mathbf{S}_x^T \quad (55)$$

Now, sub in expressions from above for \mathbf{x}_{k+1} and \mathbf{e}_{k+1} .

$$\frac{dJ}{d\mathbf{u}_{k+1}}^T = 2\mathbf{Q}_u \mathbf{u}_{k+1} + \mathbf{q}_{\delta u}^T u_j - 2\mathbf{H}_j^T \mathbf{Q}_e (\mathbf{I}e_j - \mathbf{F}_j x_j - \mathbf{G}_j u_j - \mathbf{H}_j \mathbf{u}_{k+1}) - \mathbf{H}_j^T \mathbf{q}_{\delta e}^T e_j + 2\mathbf{H}_j^T \mathbf{Q}_x (\mathbf{I}x_j + \mathbf{F}_j x_j + \mathbf{G}_j u_j + \mathbf{H}_j \mathbf{u}_{k+1}) + \mathbf{H}_j^T \mathbf{q}_{\delta x}^T x_j + \mathbf{H}_j^T \mathbf{S}_x^T \quad (56)$$

Multiply everything out

$$\frac{dJ}{d\mathbf{u}_{k+1}}^T = 2\mathbf{Q}_u \mathbf{u}_{k+1} + \mathbf{q}_{\delta u}^T u_j \quad (57)$$

$$- 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{I}e_j + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{F}_j x_j + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{G}_j u_j + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{H}_j \mathbf{u}_{k+1} - \mathbf{H}_j^T \mathbf{q}_{\delta e}^T e_j \quad (58)$$

$$+ 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{I}x_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{F}_j x_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{G}_j u_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{H}_j \mathbf{u}_{k+1} + \mathbf{H}_j^T \mathbf{q}_{\delta x}^T x_j + \mathbf{H}_j^T \mathbf{S}_x^T \quad (59)$$

Gather like terms

$$\frac{dJ}{d\mathbf{u}_{k+1}}^T = 2\mathbf{Q}_u \mathbf{u}_{k+1} + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{H}_j \mathbf{u}_{k+1} + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{H}_j \mathbf{u}_{k+1} \quad (60)$$

$$+ \mathbf{q}_{\delta u}^T u_j + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{G}_j u_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{G}_j u_j \quad (61)$$

$$- 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{I}e_j - \mathbf{H}_j^T \mathbf{q}_{\delta e}^T e_j \quad (62)$$

$$+ 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{F}_j x_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{I}x_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{F}_j x_j + \mathbf{H}_j^T \mathbf{q}_{\delta x}^T x_j \quad (63)$$

$$+ \mathbf{H}_j^T \mathbf{S}_x^T \quad (64)$$

$$= (2\mathbf{Q}_u + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{H}_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{H}_j) \mathbf{u}_{k+1} \quad (65)$$

$$+ (\mathbf{q}_{\delta u}^T + 2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{G}_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{G}_j) u_j \quad (66)$$

$$- (2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{I} + \mathbf{H}_j^T \mathbf{q}_{\delta e}^T) e_j \quad (67)$$

$$+ (2\mathbf{H}_j^T \mathbf{Q}_e \mathbf{F}_j + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{I} + 2\mathbf{H}_j^T \mathbf{Q}_x \mathbf{F}_j + \mathbf{H}_j^T \mathbf{q}_{\delta x}^T) x_j \quad (68)$$

$$+ \mathbf{H}_j^T \mathbf{S}_x^T \quad (69)$$

Set it equal to zero vector and divide by 2

$$\vec{0} = (\mathbf{Q}_u + \mathbf{H}_j^T (\mathbf{Q}_e + \mathbf{Q}_x) \mathbf{H}_j) \mathbf{u}_{k+1} \quad (70)$$

$$+ \left(\frac{1}{2} \mathbf{q}_{\delta u}^T + \mathbf{H}_j^T (\mathbf{Q}_e + \mathbf{Q}_x) \mathbf{G}_j \right) u_j \quad (71)$$

$$- \mathbf{H}_j^T \left(\mathbf{Q}_e \mathbf{I} + \frac{1}{2} \mathbf{q}_{\delta e}^T \right) e_j \quad (72)$$

$$+ \mathbf{H}_j^T \left(\mathbf{Q}_e \mathbf{F}_j + \mathbf{Q}_x \mathbf{I} + \mathbf{Q}_x \mathbf{F}_j + \frac{1}{2} \mathbf{q}_{\delta x}^T \right) x_j \quad (73)$$

$$+ \frac{1}{2} \mathbf{H}_j^T \mathbf{S}_x^T \quad (74)$$

So solve this for \mathbf{u}_{k+1}

$$\mathbf{u}_{k+1} = \mathbf{L}_u u_j + \mathbf{L}_e e_j + \mathbf{L}_x x_j + \mathbf{L}_c \quad (75)$$

$$\text{where} \quad (76)$$

$$\mathbf{L}_0 \triangleq (\mathbf{Q}_u + \mathbf{H}_j^T (\mathbf{Q}_e + \mathbf{Q}_x) \mathbf{H}_j)^{-1} \quad (77)$$

$$\mathbf{L}_u \triangleq -\mathbf{L}_0 \left(\frac{1}{2} \mathbf{q}_{\delta u}^T + \mathbf{H}_j^T (\mathbf{Q}_e + \mathbf{Q}_x) \mathbf{G}_j \right) \quad (78)$$

$$\mathbf{L}_e \triangleq \mathbf{L}_0 \mathbf{H}_j^T \left(\mathbf{Q}_e \mathbf{I} + \frac{1}{2} \mathbf{q}_{\delta e}^T \right) \quad (79)$$

$$\mathbf{L}_x \triangleq -\mathbf{L}_0 \mathbf{H}_j^T \left(\mathbf{Q}_e \mathbf{F}_j + \mathbf{Q}_x \mathbf{I} + \mathbf{Q}_x \mathbf{F}_j + \frac{1}{2} \mathbf{q}_{\delta x}^T \right) \quad (80)$$

$$\mathbf{L}_c \triangleq -\mathbf{L}_0 \frac{1}{2} \mathbf{H}_j^T \mathbf{S}_x^T \quad (81)$$