Control loops for target tracking in adaptive radiotherapy

Martin J Murphy PhD
Department of Radiation Oncology
Virginia Commonwealth University

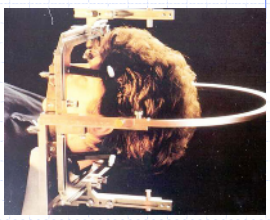
We know that many (perhaps most) target sites move internally during treatment

- The movement can occur over timescales of
  - Days (fraction to fraction changes)
  - Minutes (intra-fraction movement)
  - Seconds (respiration)

Solutions to the movable target problem

The dark ages: irradiate everything

The middle ages: Nail the patient to the treatment table



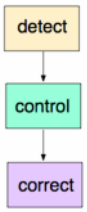
The enlightenment:

- Observe and measure the movement
- Expand the treatment volume to encompass it

The modern age: adapt to the movement

- If movement is slow enough, adaptation can be manually directed off line
- Faster changes must be automatically corrected, either on or off line
- Fastest movement (e.g., respiration) must be automatically tracked in real time
- (figure courtesy of BrainLab)

Components of a tumor tracking system



- A target localization system to measure the tumor position
- A re-alignment system to shift the beam with respect to the patient, or vice-versa
- A control loop to translate tumor position into beam/couch coordinates and synchronously reposition the beam/patient
- Error smoothing and fault detection

- The basic concepts of control and feedback in motion compensation are independent of timescale
- The control process can operate over whatever timescale is associated with the movement
- This focus of this presentation is real-time tracking but it is relevant to daily inter-fraction adaptation as well

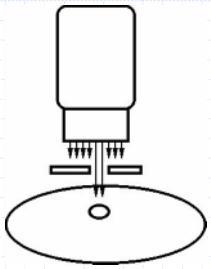
- Real-time compensation: detecting and compensating significant changes at least as fast as they occur.
- If the highest significant frequency in the motion spectrum is ω,then the real-time sampling/correcting interval should be about $1/(2\omega)$.
- Real time doesn't mean instantly.

Dynamic realignment systems

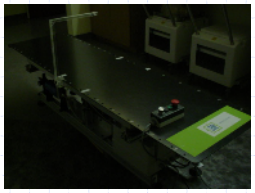
Dynamic beam shift - CyberKnife Synchrony

- X-ray imaging samples the tumor position
- surface markers correlate with the tumor position
- Optical tracking of the surface provides continuous tumor position estimates
- Real-time control loop synchronizes beam pointing with inferred breathing motion of the tumor

Dynamic beam shift - multileaf collimator



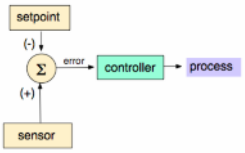
Dynamic couch shift

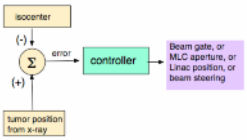


- The couch has six degrees of freedom in movement
- It is remotely controllable in real time
- The tumor position is detected and relayed to the couch
- If the tumor moves, the couch makes a compensating movement such that the tumor position remains fixed in the frame of the room
- (Images courtesy of Warren D'Souza, Univ of Maryland)

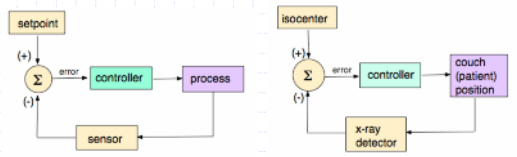
Control loops for synchronization

Open control loop (feedforward)

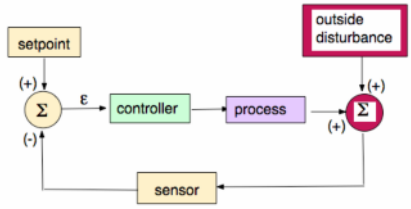




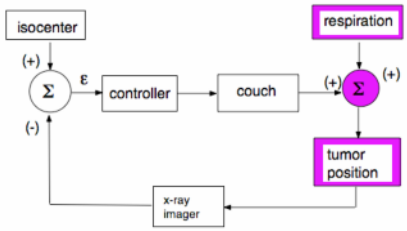
Closed control loop (feedback)



Closed loop



A closed control loop for direct tumor tracking during respiration



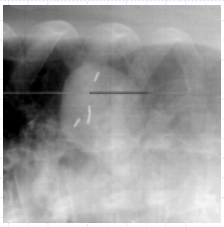
Target localization

Real-time target localization

- Follow the outline of the tumor itself in x-ray images
- Follow markers in the tumor
  - Radiographic markers
  - electromagnetic transponders
- Correlate the tumor motion with external surrogate markers

Direct tumor tracking

- Sometimes you get lucky and can clearly see the tumor outline in a 2D radiograph



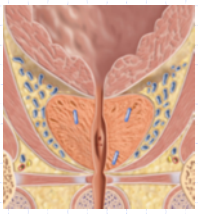
Internal fiducial tracking - when you can't clearly see the tumor itself

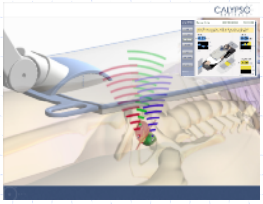
- Implant radio-opaque markers in or near the tumor before CT
- Track the fiducials fluoroscopically

Electromagnetic fiducial tracking

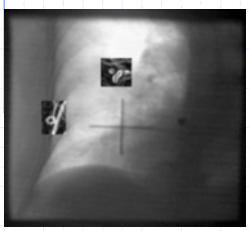
(figures courtesy of Calypso Medical Technologies, Seattle WA)

- Implant radiofrequency markers
- Track the markers electromagnetically



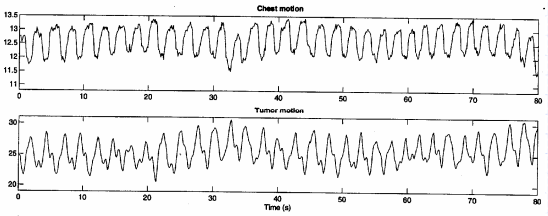


Surrogate tracking

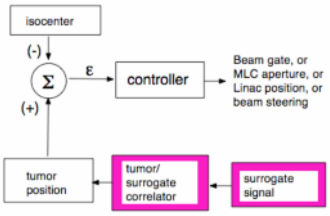


- Make a fluoroscopic video or 4D CT of the patient
- Correlate the tumor motion with the external marker motion
- During treatment, use the external marker motion to predict the tumor position

Tumor-chest correlation

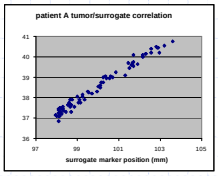


Open loop surrogate tracking



Simple tumor-chest motion correlation

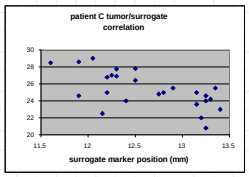
Figure 3a



- Here the correlation is linear and stationary
- Sufficient to fit a straight line:

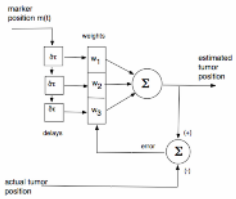
$$X_t = aX_s + b$$

Complex tumor-chest motion correlation - adaptive linear correlation



- Here the correlation changes with time
- Collect correlation points for, e.g., 30 seconds
- Fit to a straight line
- Each time a new correlation point is acquired, throw out the oldest point and redo the fit

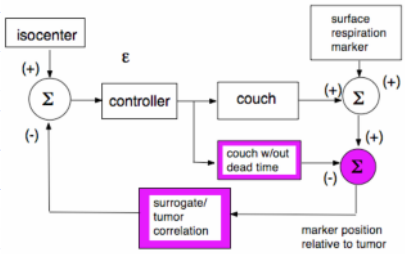
correlation via an adaptive linear filter



- Take N consecutive samples of the surrogate position
- Make a weighted sum
- Compare the weighted sum to the tumor position
- Adjust the weights to minimize the difference
- Adjust the weights with each new data sample

Closed loop surrogate tracking

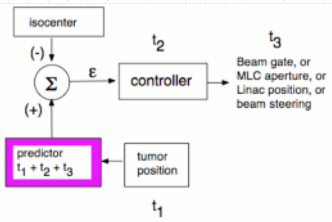
(D'Souza and McAvoy, Med Phys 33, 4701-4709, 2006)



System latency

- Nothing happens immediately - each step in the control sequence requires a finite time.
- The "system latency" must be compensated by estimating the next target position

Compensating system response times



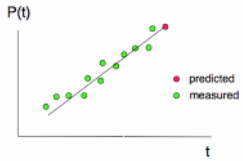
Methods of prediction

- Make a biomechanical model of the movement process
- Make a mathematical model of the movement - e.g., periodic functions for breathing
- Apply statistical methods of signal prediction
- Make a heuristic model of the movement that mimics each individual pattern

Estimating the next target position from the movement history

- Each measurement of target position is affected by two stochastic sources of noise:
  - fluctuations in the process that causes the motion - the so-called plant noise
  - fluctuations in the detection process - the measurement noise
- The prediction process must find the best estimate of the next position in the presence of these fluctuations

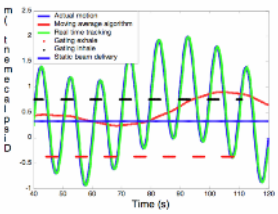
Auto-regressive prediction



- A trajectory is fitted by least squares to the prior motion data
- Each new position point is added and the trajectory is re-fitted
- This method remembers all prior data

Rapid fluctuations, slowly changing mean: weighted running average

(figure from: R George et al, Med Phys 35, 2356-2365, 2008)



$P'_{i+1} \approx \langle P_i \rangle$

$\langle P_i \rangle = (1 - w) \langle P_{i-1} \rangle + w P_i$

W is the weight for the current data point

Estimating the next target position with a Kalman filter

- The Kalman filter is a "predictor/corrector" algorithm.
- The "system" is the physical process that moves the target around.
- The filter estimates the future system state, makes a prediction of the future target position based on that estimate, compares that prediction to the measured position when it is available, uses the error in the prediction to update the system state, and repeats.

Kalman filter

X(t) is the system state at the moment t.
z(t) is the measured target position at t.
s(t+1) is the prediction of the next position.

u is the measurement noise:
R = cov(u)
w is the state (plant) noise:
Q = cov(w)

Kalman filter

- If the evolution of the system is linear, we can use a linear Kalman filter.
- This might be the case if we are tracking a tumor position from day to day while allowing for setup uncertainties and slowly-changing anatomy deformation.

linear Kalman filter

- The linear operator H relates the target position to the system state

  $z(t) = H\,X(t) + u$

- The linear operator Φ relates sequential system states

  $X(t) = \Phi\,X(t-1) + w$

How the filter works:

- **Predict**:

  Estimate the future state:

  $X'(t+1) = \Phi\,X(t)$

  Estimate the future error covariance:

  $P'(t+1) = \Phi\,P(t)\,\Phi^{T} + Q(t)$

  Estimate the next target position:

  $s(t+1) = H\,X'(t+1)$

- **Observe:** the position $z(t+1)$
- **Correct**:

  Compute the Kalman gain:

  $K(t+1) = P'(t+1)\,H^{T}\,[HP'(t+1)H^{T} + R(t+1)^{-1}]$

  Update the system state:

  $X(t+1) = X'(t+1) + K(t+1)\,[z(t+1) - s(t+1)]$

  Update the error covariance:

  $P(t+1) = P'(t+1) - K(t+1)\,H\,P'(t+1)$

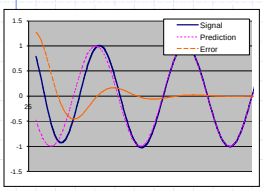
The nonlinear (extended) Kalman filter

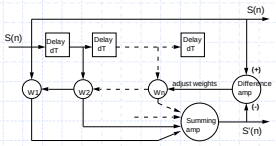
- If the changes in the system state are nonlinear, we must use an extended Kalman filter.
- This would be the case for a tumor that moves around during respiration.
- $X(t) = f(X(t-1)) + w$ (nonlinear evolution described by the function $f(X)$).
- We need a model for $f(X)$.
- To estimate the future state of the of the system we need to linearize $f(X)$ around the present state so that it can be approximated as $\Phi\,X(t)$. This can be done, e.g., via Taylor expansion.

Adaptive filters

- Adaptive filters are heuristic algorithms that mimic the incoming signal by taking sequential samples of the signal amplitude and combining them in a weighted sum to estimate the present or future amplitude
- The filters make no assumptions about the functional form of the signal or the physical mechanism producing it
- The filters can continually adjust their weighting parameters to adapt to changes in the signal shape

A basic linear adaptive filter

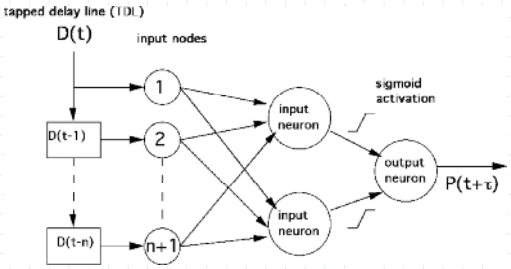




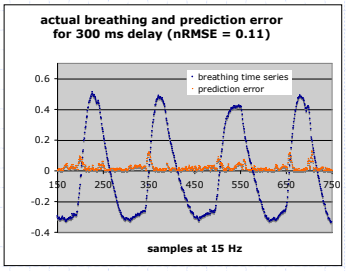
Adapting the filter

- Once the weights have been initialized, real-time position data are presented to the filter
- Each time the filter makes a position prediction for t +*T* it is recorded for later comparison to the actual position at *t* + *T*
- An error signal is generated and the weights are adjusted
- This is simply a continuation of the initial sequential training process using the new data

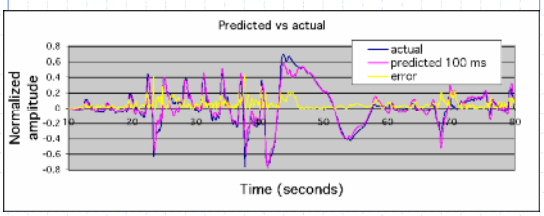
Fom linear predictor to nonlinear neural network: each neuron is a linear filter



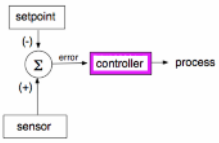
Nonlinear adaptive neural network for temporal prediction of breathing



Nonlinear adaptive neural network

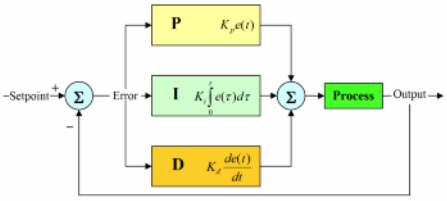


Error processing in the controller

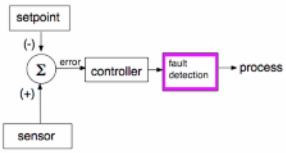


- The data comparator sends an error to the controller; the controller translates the error into a process signal:
  - Make a discrete (beam gate) or proportional response (MLC shift, couch shift)
  - Apply a threshold
  - Allow for gain in the process response
  - Filter noise from the error signal
  - Avoid instantly responding to the slightest change (response window)
  - Respond slowly to abrupt changes

Error processing - the PID (proportional-integral-differential) controller



Fault detection - test the process signal for validity



- Reachability
- Collision avoidance
- Loss of target
- Speed limits

Summary

- Image-guided adaptive radiotherapy is a system control problem
- The rate of adaptation determines the degree of automation
- Real time means as fast as necessary, not as fast as possible
- The control process can be open or closed loop
- Estimation of the next target position is a necessary part of the process
- A robust system must have error smoothing and fault detection