Uncontrolled manifold (UCM)

Jason Friedman

Tel Aviv University

14 January, 2025

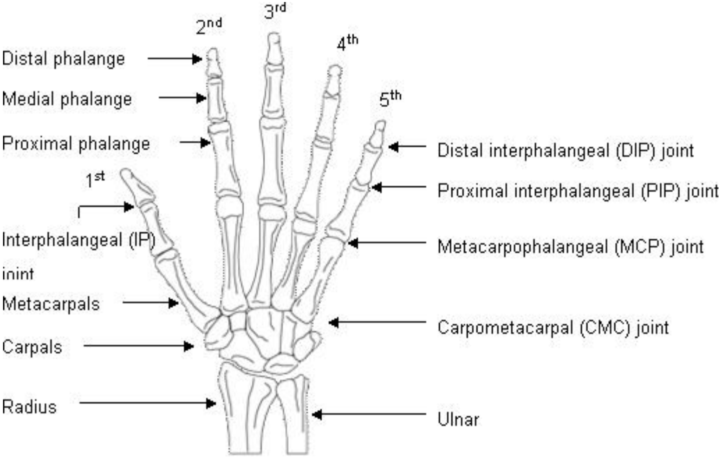
REDUNDANCY IN GRASPING

- Redundancy = more solutions than necessary
- When grasping, we are required to solve (at least) these mostly redundant problems:
  - Select grasp points for a particular task
  - Select the posture of the hand
  - Select the stiffness properties of the fingers and grasp
  - Coordinate the grip and tangential forces of the fingers
  - Share the force produced by multiple fingers
- and ideally do this in an efficient / optimal way

DEGREES OF FREEDOM PROBLEM

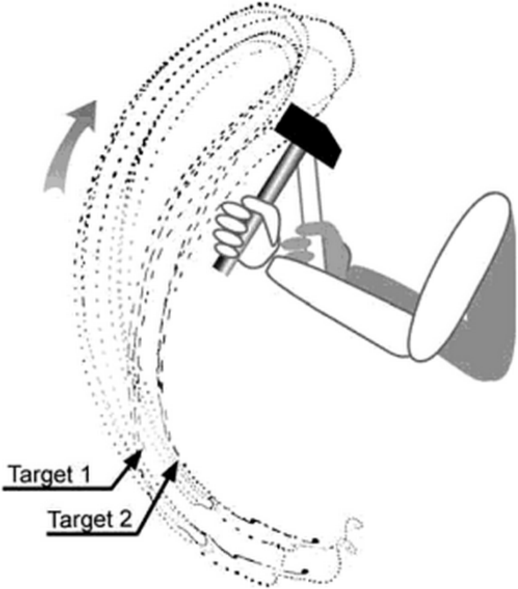
There are many degrees of freedom in the hand

> 20 joints

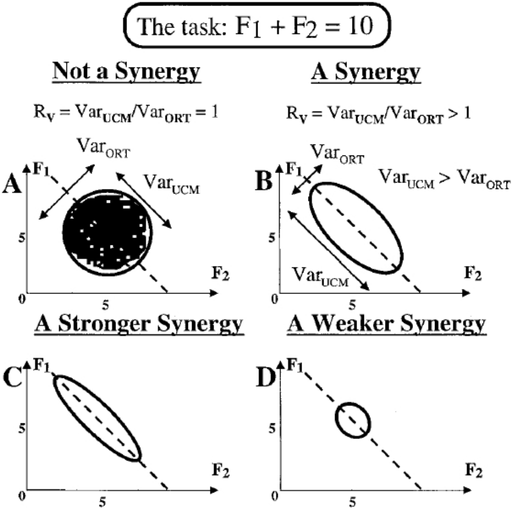


Large number of muscles

Degrees of freedom problem



Uncontrolled Manifold (UCM)



Prehension synergies

A *prehension synergy*

- is a combined change of finger forces and moments during multi-finger prehension tasks
- adjusts to changes in task parameters
- compensates for external or self-inflicted disturbances

Uncontrolled Manifold (UCM)

- The *uncontrolled manifold* approach (Scholz and Schöner, 1999) can be used to explain the variance observed when a given task has more degrees of freedom than necessary.
- We calculate which changes from the average (mean) performance that do not affect the goal of the task.
- Then the variance that does not change the performance variable (V_{UCM} - “good” variance) and variance that does change the performance variable ($V_{ORT} -$ “*bad*″*variance*) can be calculated.

CALCULATION OF THE GOOD AND BAD VARIANCE

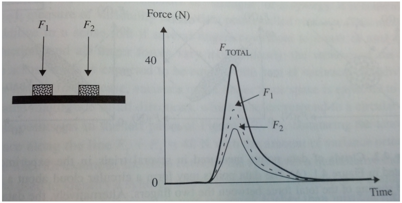
- Consider a task where the subject has to control the total force, which is the sum of the force produced by two fingers.

$$dF_{TOT} = \begin{bmatrix} 1 & 1 \end{bmatrix} d\mathbf{f}$$

where

$$d\mathbf{f} = \begin{bmatrix} df_i \\ df_m \end{bmatrix}$$

- The matrix $\begin{bmatrix} 1 & 1 \end{bmatrix}$ is the Jacobian, which transforms from a change in elemental variables to a change in performance variables



Calculation of the good and bad variance

- We now consider combinations of force produced by the fingers that do alter the total force, i.e.

$$0 = \begin{bmatrix}1 & 1\end{bmatrix} \mathbf{e_i}$$

- This is simply the null space of this transformation. We project the mean-free modes onto these directions, and sum them to find the *good* variance

$$f_{||} = \sum_{i=1}^{n-p} \left(\mathbf{e_i}^T \cdot d\mathbf{f} \right) \mathbf{e_i}$$

Calculation of the good and bad variance

- To calculate the *good* variance, V_{UCM}, we calculate the amount of this variance per degree-of-freedom (DOF)

$$V_{\mathrm{UCM}} = \frac{\sum |f_{||}|^2}{(n-p)\, N_{\mathrm{trials}}}$$

- The bad variance is calculated in a similar way:

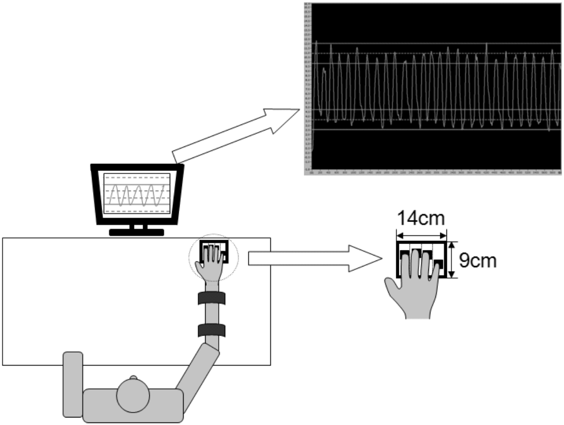
$$f_{\perp} = d\mathbf{m} - f_{||}$$

$$V_{\mathrm{ORT}} = \frac{\sum |f_{\perp}|^2}{p N_{\mathrm{trials}}}$$

Calculation of the good and bad variance

- By comparing the relative size of the good and bad variance, we can determine whether the fingers are acting as a *synergy*
- *good variance* $>$ *bad variance*: the fingers are correcting each other, so we call this a synergy
- *good variance* $\approx$ *bad variance*: we do not see a synergy
- *bad variance* $>$ *good variance*: destabilizing synergy

EXPERIMENT: EXPERIMENTAL METHODS



- The subjects had to press with all four fingers, and produce a sinusoidal force, between given targets, and with a certain timing (using a metronome)
- Feedback on the *total force* of the four fingers was shown on a monitor.

Calculation of the good and bad variance

- In this task, the subjects had to control the total force, which is just the sum of the force produced by the four fingers.
- Note that in the paper, we used modes (m) rather than forces, which are the theoretical commands given to the fingers, to take into account enslaving, but we will use finger forces for simplicity

$$dF_{TOT} = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} d\mathbf{f}$$

where

$$d\mathbf{f} = \begin{bmatrix} df_i & df_m & df_r & df_l \end{bmatrix}^T$$

MATLAB

LOAD THE DATA

- Load the data and plot it

```
data = load('data/fingerforcedata.txt');
% Take a look (first four columns are the force)
forces = data(:,1:4);
% Sample rate is 200 Hz
time = (1:size(forces,1)) .* 1/200;
plot(time,forces);
hold on;
sumforce = sum(forces,2);
plot(time,sumforce,'k');
xlabel('time (s)');
ylabel('Force (N)');
legend('Index','Middle','Ring','Little','Total');
```

Matlab

Peaks and troughs

- Find the peaks and troughs to segment the data

```
[peaks,peakLocations] = ...
  findpeaks(sumforce,'minpeakheight',20,...
                     'minpeakdistance',100);
[troughs,troughLocations] = ...
  findpeaks(-sumforce,'minpeakheight',-10,...
                      'minpeakdistance',100);
figure;
plot(time,sumforce);
hold on;
plot(time(peakLocations),...
  sumforce(peakLocations),'r*');
plot(time(troughLocations),...
  sumforce(troughLocations),'g*');
```

MATLAB

EXTRACT "UP" MOVEMENTS

- Extract the "up" movements, and time normalize to 100 samples

```
for k=2:numel(troughLocations)-1
 movementsUp{k-1} = ...
     forces(troughLocations(k):peakLocations(k),:);
 timeUp{k-1} = ...
     time(troughLocations(k):peakLocations(k));
  for m=1:4
   movementsUpCombined(k-1,m,:) = ...
    interp1(timeUp{k-1},movementsUp{k-1}(:,m),...
    linspace(timeUp{k-1}(1),timeUp{k-1}(end),100));
  end
end
```

MATLAB

EXTRACT "DOWN" MOVEMENTS

- Extract the "down" movements, and time normalize to 100 samples

```
for k=2:numel(troughLocations)-1
 movementsDown{k-1} = ...
     forces(peakLocations(k):troughLocations(k+1),:);
 timeDown{k-1} = ...
     time(peakLocations(k):troughLocations(k+1));
 for m=1:4
  movementsDownCombined(k-1,m,:) = ...
   interp1(timeDown{k-1},movementsDown{k-1}(:,m),...
   linspace(timeDown{k-1}(1),timeDown{k-1}(end),100));
 end
end
```

MATLAB

- The total force is the sum of the finger forces:

$$F_{TOT} = f_i + f_m + f_r + f_l$$

- We can compute the Jacobian using the symbolic math toolkit:

```
syms fi fm fr fl;
ftot = fi + fm + fr + fl;
J(1,1) = diff(ftot,fi);
J(1,2) = diff(ftot,fm);
J(1,3) = diff(ftot,fr);
J(1,4) = diff(ftot,fl);
```

CALCULATION OF THE GOOD AND BAD VARIANCE

- We now consider combinations of force produced by the fingers that do alter the total force, i.e.

$$\begin{aligned} 0 &= J\mathbf{f_i} \\ &= \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \mathbf{f_i} \end{aligned}$$

- This is the null space of this transformation. We use the function *null* in Matlab to find the null space

```
J = [1 1 1 1];
nullspace = null(J);
```

Calculation of the good and bad variance

- We first need to find the “mean-free” forces (i.e., subtract the mean forces)

```
numrepetitions = size(movementsUpCombined,1);
meanMovementsUp = mean(movementsUpCombined);
meanFreeMovementsUp = movementsUpCombined - ...
    repmat(meanMovementsUp,numrepetitions,1);

meanMovementsDown = mean(movementsDownCombined);
meanFreeMovementsDown = movementsDownCombined - ...
    repmat(meanMovementsDown,numrepetitions,1);
```

Calculation of the good and bad variance

- We project the mean-free forces onto these three directions (e_i), and sum them to find the *good* variance (at each time point)

$$f_{||} = \sum_{i=1}^{n-p} \left(\mathbf{e_i}^T \cdot d\mathbf{f} \right) \mathbf{e_i}$$

```
for t=1:100
  for k=1:size(nullspace,2)
    goodvariancesUp(t,k,:,:) = (nullspace(:,k)' * ...
        meanFreeMovementsUp(:,:,t)')' * ...
        nullspace(:,k)';
  end
  goodvariancesCombinedUp(:,:,t) = ...
      sum(goodvariancesUp(t,:,:,:));
end
```

Calculation of the good and bad variance

- To calculate the *good* variance, V_{UCM}, we calculate the amount of this variance per DOF

$$V_{\text{UCM}} = \frac{\sum |f_{||}|^2}{(n-p)\, N_{\text{trials}}}$$

```
vUCM_Up = squeeze(...
 sum(sum(goodvariancesCombinedUp.^2,2),1)) ...
 ./ ((4-1)*numrepetitions);
```

Calculation of the good and bad variance

- The bad variance is calculated in a similar way:

$$f_{\perp} = d\mathbf{f} - f_{||}$$

$$V_{\text{ORT}} = \frac{\sum |f_{\perp}|^2}{pN_{\text{trials}}}$$

```
badvariancesCombinedUp = meanFreeMovementsUp - ...
    goodvariancesCombinedUp;
vORT_Up = squeeze(...
  sum(sum(badvariancesCombinedUp.^2,2),1)) ...
  ./ (1*numrepetitions);
```

Comparison of good and bad variance

- We can now compare the good and bad variance
- We observe that the good variance (V_{UCM}) is approximately linearly related to the force
- We observe that the bad variance (V_{ORT}) is approximately linearly related to the force rate

Comparison of good and bad variance

- A measure of whether this is a "synergy" is given by the change in variance:

$$\Delta V = \frac{V_{UCM} - V_{ORT}}{V_{ORT}}$$

```
ΔV_Up = (vUCM_Up-vORT_Up)./vORT_Up;
```

Conclusions

- We can use the UCM method with a redundant system (i.e. more elemental variables than performance variables) to determine if there is a synergy
- This method requires that we can construct the Jacobian