Performance Bounds for Human Machine Teaming and Design

Pete Trautman
Galois Inc.
(and conversations with CSEL at Ohio State University)
Some Classes of HMT
Some Classes of HMT

- Low level shared control
  - prosthetics, wheelchairs, telepresence
  - Human, machine “share” Control
Some Classes of HMT

- Low level shared control
  - prosthetics, wheelchairs, telepresence
  - Human, machine “share” Control

- Switching control
  - Self driving cars (human OR machine; special case of “sharing”)
Some Classes of HMT

• Low level shared control
  – prosthetics, wheelchairs, telepresence
  – Human, machine “share” Control

• Switching control
  – Self driving cars (human OR machine; special case of “sharing”)

• Supervisory control
  – swarm management (human “constrains” machine)
Some Classes of HMT

• Low level shared control
  – prosthetics, wheelchairs, telepresence
  – Human, machine “share” Control

• Switching control
  – Self driving cars (human OR machine; special case of “sharing”)

• Supervisory control
  – swarm management (human “constrains” machine)

• Human Machine Team Design
  – Design of teams from high level metrics
    (Cognitive workload, performance, energy)

Mathematics of low level shared control informs all the above
Background to Approach
Robot-Crowd Model
Robot-Crowd Model

- $z_{1:t}^R = \text{platform data (not decision making!)}$
  - odometry, localization, etc
Robot-Crowd Model

- $z_{1:t}^R = \text{platform data (not decision making!)}$
  - odometry, localization, etc
- $z_{1:t}^f = \text{msmts of crowd 1, \ldots, } n_t$
Robot Crowd Model
Robot Crowd Model

• $f^R = \text{autonomy process}$
• $f = f^1, \ldots, f^{n_t} \text{ crowd process}$
Robot Crowd Model

- $f^R = \text{autonomy process}$
- $f = f^1, \ldots, f^{n_t} \text{ crowd process}$
Planning as Inference
Planning as Inference

\[ [f^R, f]^* = \arg\max_{f^R, f} p(f^R, f | \bar{z}_{1:t}) \]
Planning as Inference

\[ [f^R, f]^* = \arg\max_{f^R, f} p(f^R, f \mid \bar{z}_{1:t}) \]

\[ u(t + 1) = f^{R^*}(t + 1) \]
Planning as Inference

\[ [f^R, f]^* = \text{argmax}_{f^R, f} p(f^R, f \mid \bar{z}_{1:t}) \]

\[ u(t + 1) = f^{R*}(t + 1) \]

\[ p(f^R, f \mid \bar{z}_{1:t}) = \psi_f(f^R, f)p(f^R \mid z^R_{1:t}) \prod_{i=1}^{n_t} p(f^i \mid z^i_{1:t}) \]
Planning as Inference

\[
[f^R, f]^* = \arg\max_{f^R, f} p(f^R, f \mid \bar{z}_{1:t})
\]

\[
u(t + 1) = f^{R*}(t + 1)
\]

\[
p(f^R, f \mid \bar{z}_{1:t}) = \psi_f(f^R, f)p(f^R \mid z^R_{1:t}) \prod_{i=1}^{n_t} p(f^i \mid z^i_{1:t})
\]
Interacting Random Trajectories: Motivation
Interacting Random Trajectories:
Motivation

- $z_{1:t}^R = \text{platform data (not decision making!)}$
  - odometry, localization, etc

- $z_{1:t}^f = \text{msmts of crowd 1, \ldots, } n_t$
Interacting Random Trajectories: Motivation

- $\mathbf{z}^{R}_{1:t} = \text{platform data (not decision making!)}$
  - odometry, localization, etc

- $\mathbf{z}^{f}_{1:t} = \text{msmts of crowd 1, \ldots, } n_t$

- $\mathbf{z}^{h}_{1:t} = \text{operator input (joystick, BMI, etc)}$
Interacting Random Trajectories: Motivation

- $z_{1:t}^R = \text{platform data (not decision making!)}$
  - odometry, localization, etc
- $f^R = \text{autonomy process}$
- $z_{1:t}^f = \text{msmts of crowd } 1, \ldots, n_t$
- $z_{1:t}^h = \text{operator input (joystick, BMI, etc)}$
Interacting Random Trajectories: Motivation

- $z_{1:t}^R = \text{platform data (not decision making!)}$
  - odometry, localization, etc

- $f^R = \text{autonomy process}$

- $z_{1:t}^f = \text{msmts of crowd 1, \ldots, } n_t$

  - $f = f^1, \ldots, f^{n_t} \text{ crowd process}$

- $z_{1:t}^h = \text{operator input (joystick, BMI, etc)}$
Interacting Random Trajectories: Motivation

- $z^R_{1:t} = $ platform data (not decision making!)
  - odometry, localization, etc
  - $f^R = $ autonomy process

- $z^f_{1:t} = $ msmts of crowd 1, ..., $n_t$
  - $f = f^1, \ldots, f^{n_t}$ crowd process

- $z^h_{1:t} = $ operator input (joystick, BMI, etc)
  - $h = $ operator process
Shared Control, conceptually
Shared Control, conceptually

• “Environment” navigation couples autonomy and environment
Shared Control, conceptually

- “Environment” navigation couples autonomy and environment
- Shared control couples operator and autonomy
Shared Control, conceptually

- “Environment” navigation couples autonomy and environment
- Shared control couples operator and autonomy
- Model the relationship between operator, autonomy and environment
Interacting Random Trajectories (IRT)

\[ p(\mathbf{h}, \mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}) = \psi_h(\mathbf{h}, \mathbf{f}^R)p(\mathbf{h} \mid \mathbf{z}_{1:t}^h)p(\mathbf{f}^R, \mathbf{f} \mid \bar{\mathbf{z}}_{1:t}) \]
Interacting Random Trajectories (IRT)

\[ p(h, f^R, f \mid z_{1:t}) = \psi_h(h, f^R)p(h \mid z_{1:t}^h)p(f^R, f \mid \bar{z}_{1:t}) \]

\[ p(f^R \mid z_{1:t}^R) \]

\[ \prod_{i=1}^{n_t} p(f^i \mid z_{1:t}^i) \]
Interacting Random Trajectories (IRT)

\[ p(h, f^R, f \mid z_{1:t}) = \psi_h(h, f^R)p(h \mid z_{1:t}^h)p(f^R, f \mid \bar{z}_{1:t}) \]

\[ p(h \mid z_{1:t}^h) \]

\[ \psi_h(h, f^R) \]

\[ p(f^R \mid z_{1:t}^R) \]

\[ \psi_f(f^R, f) \]

\[ \prod_{i=1}^{n_t} p(f^i \mid z_{1:t}^i) \]
Interacting Random Trajectories (IRT)

\[ p(h, f^R, f \mid z_{1:t}) = \psi_h(h, f^R) p(h \mid z^h_{1:t}) p(f^R, f \mid \bar{z}_{1:t}) \]

\[ p(h \mid z^h_{1:t}) \]

\[ [h, f^R, f]^* = \arg \max_{h,f^R,f} p(h, f^R, f \mid z_{1:t}) \]

\[ u^s_{IRT}(t) = f^R_{t+1} \]

\[ p(f^R \mid z^R_{1:t}) \]

\[ \prod_{i=1}^{n_t} p(f^i \mid z^i_{1:t}) \]
Why focus on linear blending? De facto low level shared control architecture; see Dragan’s *A Policy Blending formalism for shared control*, *IJRR*, 2013.
Linear Blending

\[ u_{LB}^s(t) = K_h z_t^h + K_R f_t^R \]

\[ [f^R, f]^* = \arg\max_{f^R, f} p(f^R, f | \bar{z}_{1:t}) \]

- \( K_h \): “how much” control operator gets
- \( K_R \): “how much” control machine gets
- \( K_h + K_R = 1 \)

Why focus on linear blending? *De facto* low level shared control architecture; see Dragan’s *A Policy Blending formalism for shared control*, IJRR, 2013.
Lemma: IRT generalizes linear blending
Lemma: IRT generalizes linear blending

- Start with $p(h, f^R, f \mid z_{1:t}) = \psi_h(h, f^R)p(h \mid z_{1:t}^h)p(f^R, f \mid \bar{z}_{1:t})$
- Take $p(h \mid z_{1:t}^h) = \mathcal{N}(h \mid \bar{h}, \Sigma_h)$
- Take $p(f^R, f \mid \bar{z}_{1:t}) = \mathcal{N}(f^R \mid f^{R*}, \Sigma_R)$
Lemma: IRT generalizes linear blending

- Start with $p(h, f^R, f | z_{1:t}) = \psi_h(h, f^R)p(h | z_{1:t}^h)p(f^R, f | z_{1:t})$

- Take $p(h | z_{1:t}^h) = \mathcal{N}(h | \bar{h}, \Sigma_h)$

- Take $p(f^R, f | z_{1:t}) = \mathcal{N}(f^R | f^{R*}, \Sigma_R)$

Then

$$\arg\max_{h, f^R, f} p(h, f^R, f | z_{1:t}) = \Sigma(\Sigma_h^{-1}\bar{h} + \Sigma_R^{-1}f^{R*})$$

$$= K_h\bar{h} + K_Rf^{R*}$$
Lemma: IRT generalizes linear blending

- Start with \( p(h, f^R, f \mid z_{1:t}) = \psi_h(h, f^R)p(h \mid z^h_{1:t})p(f^R, f \mid \tilde{z}_{1:t}) \)
- Take \( p(h \mid z^h_{1:t}) = \mathcal{N}(h \mid \bar{h}, \Sigma_h) \)
- Take \( p(f^R, f \mid \tilde{z}_{1:t}) = \mathcal{N}(f^R \mid f^{R*}, \Sigma_R) \)

Then
\[
\arg\max_{h,f^R,f} p(h, f^R, f \mid z_{1:t}) = \Sigma(\Sigma_h^{-1}\bar{h} + \Sigma_R^{-1}f^{R*})
\]
\[
= K_h\bar{h} + K_Rf^{R*}
\]

Theorems:

1. LB suboptimal wrt operator-autonomy agreeability, safety and efficiency
2. Only optimal if operator and world are unimodal

=> Can’t use LB is ambiguity is present
Interlude: IRT as “bank” of LBs
Interlude: IRT as “bank” of LBs

Consider the following Gaussian sum approximations:

- \( p(h \mid z_{1:t}^h) = \sum_{m=1}^{N_h} \mathcal{N}(h \mid \mu_m, \Sigma_m) \)

- \( p(f^R, f \mid \bar{z}_{1:t}) = \sum_{n=1}^{N_R} \mathcal{N}(f^R \mid \mu_n, \Sigma_n) \)

- Then we have that

\[ \psi_h(h, f^R)p(h \mid z_{1:t}^h)p(f^R, f \mid \bar{z}_{1:t}) \]

\[ \approx \psi_h(h, f^R) \sum_{m=1}^{N_h} \alpha_m \mathcal{N}(h \mid \mu_m, \Sigma_m) \sum_{n=1}^{N_R} \beta_n \mathcal{N}(f^R \mid \mu_n, \Sigma_n), \]
Interlude: IRT as “bank” of LBs

Consider the following Gaussian sum approximations:

- \( p(h \mid z_{1:t}^{h}) = \sum_{m=1}^{N_{h}} N(h \mid \mu_{m}, \Sigma_{m}) \)

- \( p(f^{R}, f \mid \bar{z}_{1:t}) = \sum_{n=1}^{N_{R}} N(f^{R} \mid \mu_{n}, \Sigma_{n}) \)

- Then we have that

\[
\psi_{h}(h, f^{R}) p(h \mid z_{1:t}^{h}) p(f^{R}, f \mid \bar{z}_{1:t})
\approx \psi_{h}(h, f^{R}) \sum_{m=1}^{N_{h}} \alpha_{m} N(h \mid \mu_{m}, \Sigma_{m}) \sum_{n=1}^{N_{R}} \beta_{n} N(f^{R} \mid \mu_{n}, \Sigma_{n}),
\]

- IRT compares multiple operator/autonomy hypotheses
Interlude: IRT as “bank” of LBs

Consider the following Gaussian sum approximations:

- \( p(\mathbf{h} \mid \mathbf{z}_{1:t}^{h}) = \sum_{m=1}^{N_h} \mathcal{N}(\mathbf{h} \mid \mu_m, \Sigma_m) \)

- \( p(\mathbf{f}^R, \mathbf{f} \mid \tilde{\mathbf{z}}_{1:t}) = \sum_{n=1}^{N_R} \mathcal{N}(\mathbf{f}^R \mid \mu_n, \Sigma_n) \)

- Then we have that

\[
\psi_h(\mathbf{h}, \mathbf{f}^R)p(\mathbf{h} \mid \mathbf{z}_{1:t}^{h})p(\mathbf{f}^R, \mathbf{f} \mid \tilde{\mathbf{z}}_{1:t}) \\
\approx \psi_h(\mathbf{h}, \mathbf{f}^R) \sum_{m=1}^{N_h} \alpha_m \mathcal{N}(\mathbf{h} \mid \mu_m, \Sigma_m) \sum_{n=1}^{N_R} \beta_n \mathcal{N}(\mathbf{f}^R \mid \mu_n, \Sigma_n),
\]

- IRT compares multiple operator/autonomy hypotheses
- Searching for the most “agreeable” + safe/efficient solution
  - THEOREM: IRT optimal wrt operator-autonomy agreeability, safety, efficiency (if autonomy spans operator decision space)
HMT Design

- HMT Design: given $n$ humans (h), $m$ robots (R), and $p$ tasks (f), assign subsets of humans and robots to tasks.
HMT Design

- HMT Design: given $n$ humans (h), $m$ robots (R), and $p$ tasks (f), assign subsets of humans and robots to tasks.

Assign humans $h_1$, $h_2$ and machines $R_2$, $R_3$ to task $f_2$. 
HMT Design

- HMT Design: given $n$ humans ($h$), $m$ robots ($R$), and $p$ tasks ($f$), assign subsets of humans and robots to tasks.

- Design decisions are typically based on evaluating individual capabilities.
- Redesign happens when a Metric related to performance triggers a reallocation.
HMT Design metrics

• Consider a high level *design* metric $M(h,R,f)$ *that is based on performance*
  
  – E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  
  – Any metric $M(h,R,f)$ that does not consider shared control we call a “*heat map*” (for now)
HMT Design metrics

- Consider a high level *design* metric $M(h,R,f)$ *that is based on performance*
  - E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  - Any metric $M(h,R,f)$ that does not consider shared control we call a “*heat map*” (for now)
HMT Design metrics

• Consider a high level *design* metric $M(h,R,f)$ *that is based on performance*
  
  – E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  
  – Any metric $M(h,R,f)$ that does not consider shared control we call a “*heat map*” (for now)

\[
M(h_1,h_2,R_2,R_3,f_2) = 5
\]
HMT Design metrics

• Consider a high level \textit{design} metric $M(h,R,f)$ \textit{that is based on performance}
  – E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  – Any metric $M(h,R,f)$ that does not consider shared control we call a \textit{“heat map”} (for now)

\[ M(h_1,h_2,R_2,R_3,f_2) = 5 \]
HMT Design metrics

• Consider a high level design metric $M(h,R,f)$ that is based on performance
  – E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  – Any metric $M(h,R,f)$ that does not consider shared control we call a “heat map” (for now)

- Compute $M(h_1,h_2,R_2,R_3,f_2)$
- Compute $M(.)$ for all other subteams
- If there is a reallocation that promises to “improve performance”, do it
  • Reduce cognitive burden, reduce friction, etc.
HMT Design metrics

• Consider a high level *design* metric \( M(h,R,f) \) *that is based on performance*
  
  – E.g., Cognitive workload, team performance, free energy, friction, fit, congruence, etc.
  
  – Any metric \( M(h,R,f) \) that does not consider shared control we call a "heat map" (for now)

\[
M(h_1,h_2,R_2,R_3,f_2)=5
\]

- Compute \( M(h_1,h_2,R_2,R_3,f_2) \)
- Compute \( M(.) \) for all other subteams
- If there is a reallocation that promises to “improve performance”, do it
  - Reduce cognitive burden, reduce friction, etc
  - Math: gradient ascent on \( M(.) \)
HMT Design Corollary
HMT Design Corollary

• Corollary: linear blending (or switching control) can unnecessarily and randomly instigate “disagreement” between operator and machine.
HMT Design Corollary

• Corollary: linear blending (or switching control) can unnecessarily and randomly instigate “disagreement” between operator and machine.
HMT Design Corollary

- Corollary: linear blending (or switching control) can unnecessarily and randomly instigate “disagreement” between operator and machine.
HMT Design Corollary

- Corollary: linear blending (or switching control) can unnecessarily and randomly instigate “disagreement” between operator and machine.
HMT Design Corollary

- **Corollary:** linear blending (or switching control) can *unnecessarily and randomly* instigate “disagreement” between operator and machine.

- **Lemma:** “LB disagreement” manifests as value in $M(h, R, f)$—”LB heat”. But this heat is independent of operator and machine.
  
  — LB Heat *unnecessary and random.*
HMT Design Corollary

- **Corollary:** linear blending (or switching control) can *unnecessarily and randomly* instigate “disagreement” between operator and machine.

- **Lemma:** “LB disagreement” manifests as value in $M(h,R,f)$—”LB heat”. But this heat is independent of operator and machine.

  - LB Heat *unnecessary and random.*
Implications for HMT design
Implications for HMT design

• Theorem: Heat map is only stable for completely decoupled tasks.
Implications for HMT design

• Theorem: Heat map is only stable for completely decoupled tasks.

\[ M(h_1, h_2, R_2, R_3, f_2) = 5 \]
\[ Time = 3 \]

\[ M(h_1, h_2, R_2, R_3, f_2) = 9 \]
\[ Time = 7 \]
Theorem: Heat map is only stable for completely decoupled tasks.

Change in $M(.)$ from $5 \rightarrow 9$ could be due to shared control “heat”

Might have nothing to do with operator skill or machine skill

Redesign triggered; but nothing to fix!
Importance for HMT design
Importance for HMT design

• Theorem: Heat maps fail for design. I.e., improving $M(h,R,f)$ by redesigning team does not necessarily improve performance.
Theorem: Heat maps fail for design. I.e., improving $M(h,R,f)$ by redesigning team does not necessarily improve performance.

Lemma: With non-trivial probability, $M(h,R,f)$ triggers unnecessary redesign.
Importance for HMT design

- **Theorem:** Heat maps fail for design. I.e., improving $M(h,R,f)$ by redesigning team does not necessarily improve performance.
  - **Lemma:** With non-trivial probability, $M(h,R,f)$ triggers unnecessary redesign.
  - **Lemma:** Probability of unnecessary redesign proportional to probability of shared control friction.
Importance for HMT design

• Theorem: Heat maps fail for design. I.e., improving $M(h,R,f)$ by redesigning team does not necessarily improve performance.
  – Lemma: With non-trivial probability, $M(h,R,f)$ triggers unnecessary redesign.
  – Lemma: Probability of unnecessary redesign proportional to probability of shared control friction
  – Lemma: Probability of shared control friction grows with complexity of task
What Next if Traditional Design Fails?
What Next if Traditional Design Fails?

- Connect $M(h,R,f)$ with shared control.
What Next if Traditional Design Fails?

• Connect M(h,R,f) with shared control.

• Like constructing a building:
  – Is mortar *reliable* enough to hold two bricks together?
  – Is performance of bricks dependent on mortar?
  – If so, then we can’t design building without considering mortar
Dimensionality Mismatch

**Supervisory Control**
- 1 operator, N vehicles
- Under time pressure, supervisor can only provide direction to n<N vehicles.
- How to complete supervision?

**Prosthetics**
- N actuators to control in robot arm
- Under time pressure, user can only provide direction to n<N actuators.
- How to complete trajectory command?
Mathematics of dimensionality mismatch
Mathematics of dimensionality mismatch

\[ p(h, f^R, f | z_{1:t}) = p(f^R, f | z^R, z^f, h) p(h | z^h_{1:t}) \]
Mathematics of dimensionality mismatch

\[ p(h, f^R, f \mid z_{1:t}) = p(f^R, f \mid z^R, z^f, h)p(h \mid z^h_{1:t}) \]

*Dimensionality mismatch:*
Mathematics of dimensionality mismatch

\[ p(h, f^R, f \mid z_{1:t}) = p(f^R, f \mid z^R, z^f, h)p(h \mid z^h_{1:t}) \]

**Dimensionality mismatch:**

\[ p(h \mid z^h_{1:t}) \approx \sum_{i=1}^{n} w_i \delta(h = h_i) \]

\[ \Rightarrow \]

\[ p(h, f^R, f \mid z_{1:t}) \approx p(f^R, f \mid z^R, z^f, h) \sum_{i=1}^{n} w_i \delta(h = h_i) \]
Mathematics of dimensionality mismatch

\[ p(h, f^R, f \mid z_{1:t}) = p(f^R, f \mid z^R, z^f, h)p(h \mid z^h_{1:t}) \]

**Dimensionality mismatch:**

\[ p(h \mid z^h_{1:t}) \approx \sum_{i=1}^{n} w_i \delta(h = h_i) \]

\[\Rightarrow\]

\[ p(h, f^R, f \mid z_{1:t}) \approx p(f^R, f \mid z^R, z^f, h) \sum_{i=1}^{n} w_i \delta(h = h_i) \]

*How do we make \( n \to N \)?*
Lower Bounding Teams

Performance vs. Teaming Stress

- Human Performance
- Machine Performance
- Team Performance

Teaming Stress (environment complexity, poor comms, etc.)
Lower Bounding Teams

Motivation
Lower Bounding Teams

Motivation
- Commonplace failure of existing HMT architectures (linear blending, function allocation,...)
Lower Bounding Teams

Motivation
- Commonplace failure of existing HMT architectures (linear blending, function allocation, ...)
- Why do teaming if this property doesn’t hold?
Lower Bounding Teams

Motivation
- Commonplace failure of existing HMT architectures (linear blending, function allocation,...)
- Why do teaming if this property doesn’t hold?

Invariant to:
Lower Bounding Teams

Motivation

- Commonplace failure of existing HMT architectures (linear blending, function allocation,...)
- Why do teaming if this property doesn’t hold?

Invariant to:
Human modeling fidelity, human decision making, ...
**Motivation**
- Commonplace failure of existing HMT architectures (linear blending, function allocation, ...)
- Why do teaming if this property doesn’t hold?

**Invariant to:**
Human modeling fidelity, human decision making, ...

-Evidence of robustness to model fidelity
**Motivation**
- Commonplace failure of existing HMT architectures (linear blending, function allocation, ...)
- Why do teaming if this property doesn’t hold?

**Invariant to:**
Human modeling fidelity, human decision making, ...
- Evidence of robustness to model fidelity
- Evidence that meeting lower bound will enable *exceeding lower bound (multiplicative teams)*
Lower Bounding Teams

Motivation
• Commonplace failure of existing HMT architectures (linear blending, function allocation,...)
• Why do teaming if this property doesn’t hold?

Invariant to:
Human modeling fidelity, human decision making, ...
- Evidence of robustness to model fidelity
- Evidence that meeting lower bound will enable exceeding lower bound (multiplicative teams)
- Need new representation that takes “interaction” as basic unit