# Autonomous Robots

## Disambiguation of Human Intent Through Control Space Selection

--Manuscript Draft--

| | |
|---|---|
| **Manuscript Number:** | |
| **Full Title:** | Disambiguation of Human Intent Through Control Space Selection |
| **Article Type:** | S.I. : RSS 2017 (by invitation only) |
| **Keywords:** | Shared Autonomy; Intent Inference; Intent Disambiguation; Assistive Robotics. |
| **Corresponding Author:** | Deepak Edakkattil Gopinath, M.S<br>Northwestern University<br>Evanston, IL UNITED STATES |
| **Corresponding Author Secondary Information:** | |
| **Corresponding Author's Institution:** | Northwestern University |
| **Corresponding Author's Secondary Institution:** | |
| **First Author:** | Deepak E. Gopinath, M.S |
| **First Author Secondary Information:** | |
| **Order of Authors:** | Deepak E. Gopinath, M.S |
| | Brenna D Argall, Ph.D |
| **Order of Authors Secondary Information:** | |
| **Funding Information:** | National Science Foundation (1544741) — Dr. Brenna D Argall |
| **Abstract:** | Assistive shared-control robots have the potential to transform the lives of millions of people afflicted with severe motor impairments as a result of spinal cord or brain injuries. The effectiveness and usefulness of shared-control robots is closely related to their ability to infer the user's needs and intentions and is often a limiting factor for providing appropriate assistance quickly, confidently and accurately. The contributions of this paper are three-fold: first, we propose a goal disambiguation algorithm which enhances the intent inference and assistive capabilities of a shared-control assistive robotic arm. Second, we introduce a novel intent inference algorithm that works in conjunction with the disambiguation scheme, inspired by dynamic field theory in which the time evolution of the probability distribution over goals is specified as a dynamical system. Third, we present a pilot human subject study to evaluate the efficacy of the disambiguation system. This study was performed with eight subjects. Our results suggest that (a) the disambiguation system has a greater utility value as the control interface becomes more limited and the task becomes more complex, (b) subjects demonstrated a diverse range of disambiguation request behavior with a greater concentration in the earlier parts of the trial and (c) there are no differences in the onset of robot assistance between different mode switching paradigms across tasks or across interfaces. |
| **Suggested Reviewers:** | Dorsa Sadigh<br>Stanford University<br>dorsa@cs.stanford.edu |
| | Sarah Sebo<br>Yale University<br>sarah.sebo@yale.edu |
| | Stefanos Nikolaidis<br>Carnegie Mellon University<br>snikolai@andrew.cmu.edu |
| | Rachel Holladay<br>Massachusetts Institute of Technology |

| | |
|---|---|
| | rhollada@mit.edu |

```
This is pdfTeX, Version 3.14159265-2.6-1.40.16 (TeX Live 2015/W32TeX)
(preloaded format=pdflatex 2016.4.4)  1 DEC 2017 20:38
entering extended mode
 restricted \write18 enabled.
 %&-line parsing enabled.
**./gopinathargallauro2017.tex
(./gopinathargallauro2017.tex
LaTeX2e <2016/03/31>
Babel <3.9q> and hyphenation patterns for 81 language(s) loaded.
(c:/TeXLive/2015/texmf-dist/tex/latex/base/fix-cm.sty
Package: fix-cm 2015/01/14 v1.1t fixes to LaTeX
(c:/TeXLive/2015/texmf-dist/tex/latex/base/ts1enc.def
File: ts1enc.def 2001/06/05 v3.0e (jk/car/fm) Standard LaTeX file
)) (./svjour3.cls
Document Class: svjour3 2007/05/08 v3.2
LaTeX document class for Springer journals
(c:/TeXLive/2015/texmf-dist/tex/latex/base/fleqn.clo
File: fleqn.clo 2015/03/31 v1.1i Standard LaTeX option (flush left
equations)
\mathindent=\dimen102
Applying: [2015/01/01] Make \[ robust on input line 50.
LaTeX Info: Redefining \[ on input line 51.
Already applied: [0000/00/00] Make \[ robust on input line 62.
Applying: [2015/01/01] Make \] robust on input line 74.
LaTeX Info: Redefining \] on input line 75.
Already applied: [0000/00/00] Make \] robust on input line 83.
)
Class Springer-SVJour3 Info: extra/valid Springer sub-package (-> *.clo)
(Springer-SVJour3)           not found in option list of \documentclass
(Springer-SVJour3)           - autoactivating "global" style.
(./svglov3.clo
File: svglov3.clo 2009/12/18 v3.2 style option for standardised journals
SVJour Class option: svglov3.clo for standardised journals
)
LaTeX Font Info:    Redeclaring math symbol \Gamma on input line 147.
LaTeX Font Info:    Redeclaring math symbol \Delta on input line 148.
LaTeX Font Info:    Redeclaring math symbol \Theta on input line 149.
LaTeX Font Info:    Redeclaring math symbol \Lambda on input line 150.
LaTeX Font Info:    Redeclaring math symbol \Xi on input line 151.
LaTeX Font Info:    Redeclaring math symbol \Pi on input line 152.
LaTeX Font Info:    Redeclaring math symbol \Sigma on input line 153.
LaTeX Font Info:    Redeclaring math symbol \Upsilon on input line 154.
LaTeX Font Info:    Redeclaring math symbol \Phi on input line 155.
LaTeX Font Info:    Redeclaring math symbol \Psi on input line 156.
LaTeX Font Info:    Redeclaring math symbol \Omega on input line 157.
\logodepth=\dimen103
\headerboxheight=\dimen104
\betweenumberspace=\dimen105
\aftertext=\dimen106
\headlineindent=\dimen107
\c@inst=\count79
\c@auth=\count80
\instindent=\dimen108
\authrun=\box26
```

```
\authorrunning=\toks14
\titrun=\box27
\titlerunning=\toks15
\combirun=\box28
\c@lastpage=\count81
\rubricwidth=\dimen109
\c@section=\count82
\c@subsection=\count83
\c@subsubsection=\count84
\c@paragraph=\count85
\c@subparagraph=\count86
\spthmsep=\dimen110
\c@theorem=\count87
\c@case=\count88
\c@conjecture=\count89
\c@corollary=\count90
\c@definition=\count91
\c@example=\count92
\c@exercise=\count93
\c@lemma=\count94
\c@note=\count95
\c@problem=\count96
\c@property=\count97
\c@proposition=\count98
\c@question=\count99
\c@solution=\count100
\c@remark=\count101
\c@figure=\count102
\c@table=\count103
\abovecaptionskip=\skip41
\belowcaptionskip=\skip42
\figcapgap=\dimen111
\tabcapgap=\dimen112
\figgap=\dimen113
\bibindent=\dimen114
\@tempcntc=\count104
) (c:/TeXLive/2015/texmf-dist/tex/latex/graphics/graphicx.sty
Package: graphicx 2014/10/28 v1.0g Enhanced LaTeX Graphics (DPC,SPQR)
(c:/TeXLive/2015/texmf-dist/tex/latex/graphics/keyval.sty
Package: keyval 2014/10/28 v1.15 key=value parser (DPC)
\KV@toks@=\toks16
) (c:/TeXLive/2015/texmf-dist/tex/latex/graphics/graphics.sty
Package: graphics 2016/01/03 v1.0q Standard LaTeX Graphics (DPC,SPQR)
(c:/TeXLive/2015/texmf-dist/tex/latex/graphics/trig.sty
Package: trig 2016/01/03 v1.10 sin cos tan (DPC)
) (c:/TeXLive/2015/texmf-dist/tex/latex/latexconfig/graphics.cfg
File: graphics.cfg 2010/04/23 v1.9 graphics configuration of TeX Live
)
Package graphics Info: Driver file: pdftex.def on input line 95.
(c:/TeXLive/2015/texmf-dist/tex/latex/pdftex-def/pdftex.def
File: pdftex.def 2011/05/27 v0.06d Graphics/color for pdfTeX
(c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/infwarerr.sty
Package: infwarerr 2010/04/08 v1.3 Providing info/warning/error messages
(HO)
```

```
) (c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/ltxcmds.sty
Package: ltxcmds 2011/11/09 v1.22 LaTeX kernel commands for general use
(HO)
)
\Gread@gobject=\count105
))
\Gin@req@height=\dimen115
\Gin@req@width=\dimen116
) (c:/TeXLive/2015/texmf-dist/tex/latex/amsfonts/amssymb.sty
Package: amssymb 2013/01/14 v3.01 AMS font symbols
(c:/TeXLive/2015/texmf-dist/tex/latex/amsfonts/amsfonts.sty
Package: amsfonts 2013/01/14 v3.01 Basic AMSFonts support
\@emptytoks=\toks17
\symAMSa=\mathgroup4
\symAMSb=\mathgroup5
LaTeX Font Info:    Overwriting math alphabet `\mathfrak' in version
`bold'
(Font)                  U/euf/m/n --> U/euf/b/n on input line 106.
)) (c:/TeXLive/2015/texmf-dist/tex/latex/amsmath/amsmath.sty
Package: amsmath 2016/03/10 v2.15b AMS math features
\@mathmargin=\skip43
For additional information on amsmath, use the `?' option.
(c:/TeXLive/2015/texmf-dist/tex/latex/amsmath/amstext.sty
Package: amstext 2000/06/29 v2.01 AMS text
(c:/TeXLive/2015/texmf-dist/tex/latex/amsmath/amsgen.sty
File: amsgen.sty 1999/11/30 v2.0 generic functions
\@emptytoks=\toks18
\ex@=\dimen117
)) (c:/TeXLive/2015/texmf-dist/tex/latex/amsmath/amsbsy.sty
Package: amsbsy 1999/11/29 v1.2d Bold Symbols
\pmbraise@=\dimen118
) (c:/TeXLive/2015/texmf-dist/tex/latex/amsmath/amsopn.sty
Package: amsopn 2016/03/08 v2.02 operator names
)
\inf@bad=\count106
LaTeX Info: Redefining \frac on input line 199.
\uproot@=\count107
\leftroot@=\count108
LaTeX Info: Redefining \overline on input line 297.
\classnum@=\count109
\DOTSCASE@=\count110
LaTeX Info: Redefining \ldots on input line 394.
LaTeX Info: Redefining \dots on input line 397.
LaTeX Info: Redefining \cdots on input line 518.
\Mathstrutbox@=\box29
\strutbox@=\box30
\big@size=\dimen119
LaTeX Font Info:    Redeclaring font encoding OML on input line 634.
LaTeX Font Info:    Redeclaring font encoding OMS on input line 635.

Package amsmath Warning: Unable to redefine math accent \vec.

\macc@depth=\count111
\c@MaxMatrixCols=\count112
```

```
\dotsspace@=\muskip10
\c@parentequation=\count113
\dspbrk@lvl=\count114
\tag@help=\toks19
\row@=\count115
\column@=\count116
\maxfields@=\count117
\andhelp@=\toks20
\eqnshift@=\dimen120
\alignsep@=\dimen121
\tagshift@=\dimen122
\tagwidth@=\dimen123
\totwidth@=\dimen124
\lineht@=\dimen125
\@envbody=\toks21
\multlinegap=\skip44
\multlinetaggap=\skip45
\mathdisplay@stack=\toks22
LaTeX Info: Redefining \[ on input line 2739.
LaTeX Info: Redefining \] on input line 2740.
) (c:/TeXLive/2015/texmf-dist/tex/latex/base/latexsym.sty
Package: latexsym 1998/08/17 v2.2e Standard LaTeX package (lasy symbols)
\symlasy=\mathgroup6
LaTeX Font Info:    Overwriting symbol font `lasy' in version `bold'
(Font)                  U/lasy/m/n --> U/lasy/b/n on input line 52.
) (c:/TeXLive/2015/texmf-dist/tex/latex/float/float.sty
Package: float 2001/11/08 v1.3d Float enhancements (AL)
\c@float@type=\count118
\float@exts=\toks23
\float@box=\box31
\@float@everytoks=\toks24
\@floatcapt=\box32
) (c:/TeXLive/2015/texmf-dist/tex/latex/graphics/epsfig.sty
Package: epsfig 1999/02/16 v1.7a (e)psfig emulation (SPQR)
\epsfxsize=\dimen126
\epsfysize=\dimen127
) (c:/TeXLive/2015/texmf-dist/tex/latex/subfigure/subfigure.sty
Package: subfigure 2002/03/15 v2.1.5 subfigure package
\subfigtopskip=\skip46
\subfigcapskip=\skip47
\subfigcaptopadj=\dimen128
\subfigbottomskip=\skip48
\subfigcapmargin=\dimen129
\subfiglabelskip=\skip49
\c@subfigure=\count119
\c@lofdepth=\count120
\c@subtable=\count121
\c@lotdepth=\count122
****************************************
* Local config file subfigure.cfg used *
****************************************
(c:/TeXLive/2015/texmf-dist/tex/latex/subfigure/subfigure.cfg)
\subfig@top=\skip50
\subfig@bottom=\skip51
```

```
) (c:/TeXLive/2015/texmf-dist/tex/latex/mathtools/mathtools.sty
Package: mathtools 2015/11/12 v1.18 mathematical typesetting tools
(c:/TeXLive/2015/texmf-dist/tex/latex/tools/calc.sty
Package: calc 2014/10/28 v4.3 Infix arithmetic (KKT,FJ)
\calc@Acount=\count123
\calc@Bcount=\count124
\calc@Adimen=\dimen130
\calc@Bdimen=\dimen131
\calc@Askip=\skip52
\calc@Bskip=\skip53
LaTeX Info: Redefining \setlength on input line 80.
LaTeX Info: Redefining \addtolength on input line 81.
\calc@Ccount=\count125
\calc@Cskip=\skip54
) (c:/TeXLive/2015/texmf-dist/tex/latex/mathtools/mhsetup.sty
Package: mhsetup 2010/01/21 v1.2a programming setup (MH)
)
LaTeX Info: Thecontrolsequence`\('isalreadyrobust on input line 129.
LaTeX Info: Thecontrolsequence`\)'isalreadyrobust on input line 129.
LaTeX Info: Thecontrolsequence`\['isalreadyrobust on input line 129.
LaTeX Info: Thecontrolsequence`\]'isalreadyrobust on input line 129.
\g_MT_multlinerow_int=\count126
\l_MT_multwidth_dim=\dimen132
\origjot=\skip55
\l_MT_shortvdotswithinadjustabove_dim=\dimen133
\l_MT_shortvdotswithinadjustbelow_dim=\dimen134
\l_MT_above_intertext_sep=\dimen135
\l_MT_below_intertext_sep=\dimen136
\l_MT_above_shortintertext_sep=\dimen137
\l_MT_below_shortintertext_sep=\dimen138
) (c:/TeXLive/2015/texmf-dist/tex/latex/bbm-macros/bbm.sty
Package: bbm 1999/03/15 V 1.2 provides fonts for set symbols - TH
LaTeX Font Info:    Overwriting math alphabet `\mathbbm' in version
`bold'
(Font)                  U/bbm/m/n --> U/bbm/bx/n on input line 33.
LaTeX Font Info:    Overwriting math alphabet `\mathbbmss' in version
`bold'
(Font)                  U/bbmss/m/n --> U/bbmss/bx/n on input line 35.
) (c:/TeXLive/2015/texmf-dist/tex/latex/lipsum/lipsum.sty
Package: lipsum 2014/07/27 v1.3 150 paragraphs of Lorem Ipsum dummy text
\c@lips@count=\count127
) (c:/TeXLive/2015/texmf-dist/tex/latex/adjustbox/adjustbox.sty
Package: adjustbox 2012/05/21 v1.0 Adjusting TeX boxes (trim, clip, ...)
(c:/TeXLive/2015/texmf-dist/tex/latex/xkeyval/xkeyval.sty
Package: xkeyval 2014/12/03 v2.7a package option processing (HA)
(c:/TeXLive/2015/texmf-dist/tex/generic/xkeyval/xkeyval.tex
(c:/TeXLive/2015/te
xmf-dist/tex/generic/xkeyval/xkvutils.tex
\XKV@toks=\toks25
\XKV@tempa@toks=\toks26
)
\XKV@depth=\count128
File: xkeyval.tex 2014/12/03 v2.7a key=value parser (HA)
)) (c:/TeXLive/2015/texmf-dist/tex/latex/adjustbox/adjcalc.sty
```

```
Package: adjcalc 2012/05/16 v1.1 Provides advanced setlength with
multiple back
-ends (calc, etex, pgfmath)
) (c:/TeXLive/2015/texmf-dist/tex/latex/adjustbox/trimclip.sty
Package: trimclip 2012/05/16 v1.0 Trim and clip general TeX material
(c:/TeXLive/2015/texmf-dist/tex/latex/collectbox/collectbox.sty
Package: collectbox 2012/05/17 v0.4b Collect macro arguments as boxes
\collectedbox=\box33
)
\tc@llx=\dimen139
\tc@lly=\dimen140
\tc@urx=\dimen141
\tc@ury=\dimen142
Package trimclip Info: Using driver 'tc-pdftex.def'.
(c:/TeXLive/2015/texmf-dist/tex/latex/adjustbox/tc-pdftex.def
File: tc-pdftex.def 2012/05/13 v1.0 Clipping driver for pdftex
))
\adjbox@Width=\dimen143
\adjbox@Height=\dimen144
\adjbox@Depth=\dimen145
\adjbox@Totalheight=\dimen146
(c:/TeXLive/2015/texmf-dist/tex/latex/ifoddpage/ifoddpage.sty
Package: ifoddpage 2011/09/13 v1.0 Conditionals for odd/even page
detection
\c@checkoddpage=\count129
) (c:/TeXLive/2015/texmf-dist/tex/latex/varwidth/varwidth.sty
Package: varwidth 2009/03/30 ver 0.92;  Variable-width minipages
\@vwid@box=\box34
\sift@deathcycles=\count130
\@vwid@loff=\dimen147
\@vwid@roff=\dimen148
)) (c:/TeXLive/2015/texmf-dist/tex/generic/ulem/ulem.sty
\UL@box=\box35
\UL@hyphenbox=\box36
\UL@skip=\skip56
\UL@hook=\toks27
\UL@height=\dimen149
\UL@pe=\count131
\UL@pixel=\dimen150
\ULC@box=\box37
Package: ulem 2012/05/18
\ULdepth=\dimen151
) (c:/TeXLive/2015/texmf-dist/tex/latex/wrapfig/wrapfig.sty
\wrapoverhang=\dimen152
\WF@size=\dimen153
\c@WF@wrappedlines=\count132
\WF@box=\box38
\WF@everypar=\toks28
Package: wrapfig 2003/01/31  v 3.6
) (c:/TeXLive/2015/texmf-dist/tex/latex/multirow/multirow.sty
\bigstrutjot=\dimen154
) (c:/TeXLive/2015/texmf-dist/tex/latex/preprint/balance.sty
Package: balance 1999/02/23 4.3 (PWD)
\oldvsize=\dimen155
```

```
) (c:/TeXLive/2015/texmf-dist/tex/latex/graphics/color.sty
Package: color 2016/01/03 v1.1b Standard LaTeX Color (DPC)
(c:/TeXLive/2015/texmf-dist/tex/latex/latexconfig/color.cfg
File: color.cfg 2007/01/18 v1.5 color configuration of teTeX/TeXLive
)
Package color Info: Driver file: pdftex.def on input line 143.
) (c:/TeXLive/2015/texmf-dist/tex/latex/url/url.sty
\Urlmuskip=\muskip11
Package: url 2013/09/16  ver 3.4  Verb mode for urls, etc.
) (./breakcites.sty) (c:/TeXLive/2015/texmf-
dist/tex/latex/microtype/microtype.
sty
Package: microtype 2013/05/23 v2.5a Micro-typographical refinements (RS)
\MT@toks=\toks29
\MT@count=\count133
LaTeX Info: Redefining \textls on input line 766.
\MT@outer@kern=\dimen156
LaTeX Info: Redefining \textmicrotypecontext on input line 1285.
\MT@listname@count=\count134
(c:/TeXLive/2015/texmf-dist/tex/latex/microtype/microtype-pdftex.def
File: microtype-pdftex.def 2013/05/23 v2.5a Definitions specific to
pdftex (RS)

LaTeX Info: Redefining \lsstyle on input line 915.
LaTeX Info: Redefining \lslig on input line 915.
\MT@outer@space=\skip57
)
Package microtype Info: Loading configuration file microtype.cfg.
(c:/TeXLive/2015/texmf-dist/tex/latex/microtype/microtype.cfg
File: microtype.cfg 2013/05/23 v2.5a microtype main configuration file
(RS)
! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.64    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.64    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.64    }

I'm ignoring this; it doesn't match any \if.
```

```
! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.64    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.64    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.71    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.71    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.71    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.71    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.71    }
```

```
I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.81    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.81    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.81    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.81    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.81    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                  \def #3{#6}\expandafter
\e...
l.86    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
```

```
                                                    \def #3{#6}\expandafter
\e...
l.86    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.86    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.86    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.86    }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.89    { font = */*/*/*/* }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.89    { font = */*/*/*/* }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.89    { font = */*/*/*/* }

I'm ignoring this; it doesn't match any \if.
```

```
! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.89    { font = */*/*/*/* }

I'm ignoring this; it doesn't match any \if.

! Extra \else.
\XKV@wh@list ...r \expandafter \XKV@wh@list \else
                                                    \def #3{#6}\expandafter
\e...
l.89    { font = */*/*/*/* }

I'm ignoring this; it doesn't match any \if.

)) (./natbib.sty
Package: natbib 1998/07/14 6.8c (PWD)
\bibhang=\skip58
\bibsep=\skip59
LaTeX Info: Redefining \cite on input line 467.
\c@NAT@ctr=\count135
) (c:/TeXLive/2015/texmf-dist/tex/latex/algorithms/algorithm.sty
Package: algorithm 2009/08/24 v0.1 Document Style `algorithm' - floating
enviro
nment
(c:/TeXLive/2015/texmf-dist/tex/latex/base/ifthen.sty
Package: ifthen 2014/09/29 v1.1c Standard LaTeX ifthen package (DPC)
)
\@float@every@algorithm=\toks30
\c@algorithm=\count136
) (c:/TeXLive/2015/texmf-dist/tex/latex/algorithms/algorithmic.sty
Package: algorithmic 2009/08/24 v0.1 Document Style `algorithmic'
\c@ALC@unique=\count137
\c@ALC@line=\count138
\c@ALC@rem=\count139
\c@ALC@depth=\count140
\ALC@tlm=\skip60
\algorithmicindent=\skip61
) (c:/TeXLive/2015/texmf-dist/tex/latex/breqn/breqn.sty
(c:/TeXLive/2015/texmf-
dist/tex/latex/l3kernel/expl3.sty
Package: expl3 2016/03/28 v6468 L3 programming layer (loader)
(c:/TeXLive/2015/texmf-dist/tex/latex/l3kernel/expl3-code.tex
Package: expl3 2016/03/28 v6468 L3 programming layer (code)
L3 Module: l3bootstrap 2016/02/12 v6412 L3 Bootstrap code
L3 Module: l3names 2016/03/11 v6433 L3 Namespace for primitives
L3 Module: l3basics 2015/11/22 v6315 L3 Basic definitions
L3 Module: l3expan 2015/09/10 v5983 L3 Argument expansion
L3 Module: l3tl 2016/03/26 v6465 L3 Token lists
L3 Module: l3str 2016/03/24 v6441 L3 Strings
L3 Module: l3seq 2015/08/05 v5777 L3 Sequences and stacks
L3 Module: l3int 2016/03/24 v6441 L3 Integers
\c_max_int=\count141
```

```
\l_tmpa_int=\count142
\l_tmpb_int=\count143
\g_tmpa_int=\count144
\g_tmpb_int=\count145
L3 Module: l3quark 2015/08/17 v5855 L3 Quarks
L3 Module: l3prg 2015/11/01 v6216 L3 Control structures
\g__prg_map_int=\count146
L3 Module: l3clist 2015/09/02 v5901 L3 Comma separated lists
L3 Module: l3token 2016/03/26 v6465 L3 Experimental token manipulation
L3 Module: l3prop 2016/01/05 v6366 L3 Property lists
L3 Module: l3msg 2016/03/26 v6464 L3 Messages
L3 Module: l3file 2016/03/25 v6458 L3 File and I/O operations
\l_iow_line_count_int=\count147
\l__iow_target_count_int=\count148
\l__iow_current_line_int=\count149
\l__iow_current_word_int=\count150
\l__iow_current_indentation_int=\count151
L3 Module: l3skip 2016/01/05 v6366 L3 Dimensions and skips
\c_zero_dim=\dimen157
\c_max_dim=\dimen158
\l_tmpa_dim=\dimen159
\l_tmpb_dim=\dimen160
\g_tmpa_dim=\dimen161
\g_tmpb_dim=\dimen162
\c_zero_skip=\skip62
\c_max_skip=\skip63
\l_tmpa_skip=\skip64
\l_tmpb_skip=\skip65
\g_tmpa_skip=\skip66
\g_tmpb_skip=\skip67
\c_zero_muskip=\muskip12
\c_max_muskip=\muskip13
\l_tmpa_muskip=\muskip14
\l_tmpb_muskip=\muskip15
\g_tmpa_muskip=\muskip16
\g_tmpb_muskip=\muskip17
L3 Module: l3keys 2015/11/17 v6284 L3 Key-value interfaces
\g__keyval_level_int=\count152
\l_keys_choice_int=\count153
L3 Module: l3fp 2016/03/26 v6465 L3 Floating points
\c__fp_leading_shift_int=\count154
\c__fp_middle_shift_int=\count155
\c__fp_trailing_shift_int=\count156
\c__fp_big_leading_shift_int=\count157
\c__fp_big_middle_shift_int=\count158
\c__fp_big_trailing_shift_int=\count159
\c__fp_Bigg_leading_shift_int=\count160
\c__fp_Bigg_middle_shift_int=\count161
\c__fp_Bigg_trailing_shift_int=\count162
L3 Module: l3box 2015/08/09 v5822 L3 Experimental boxes
\c_empty_box=\box39
\l_tmpa_box=\box40
\l_tmpb_box=\box41
\g_tmpa_box=\box42
```

```
\g_tmpb_box=\box43
L3 Module: l3coffins 2016/03/24 v6440 L3 Coffin code layer
\l__coffin_internal_box=\box44
\l__coffin_internal_dim=\dimen163
\l__coffin_offset_x_dim=\dimen164
\l__coffin_offset_y_dim=\dimen165
\l__coffin_x_dim=\dimen166
\l__coffin_y_dim=\dimen167
\l__coffin_x_prime_dim=\dimen168
\l__coffin_y_prime_dim=\dimen169
\c_empty_coffin=\box45
\l__coffin_aligned_coffin=\box46
\l__coffin_aligned_internal_coffin=\box47
\l_tmpa_coffin=\box48
\l_tmpb_coffin=\box49
\l__coffin_display_coffin=\box50
\l__coffin_display_coord_coffin=\box51
\l__coffin_display_pole_coffin=\box52
\l__coffin_display_offset_dim=\dimen170
\l__coffin_display_x_dim=\dimen171
\l__coffin_display_y_dim=\dimen172
L3 Module: l3color 2014/08/23 v5354 L3 Experimental color support
L3 Module: l3sys 2015/09/25 v6087 L3 Experimental system/runtime
functions
L3 Module: l3candidates 2016/03/25 v6456 L3 Experimental additions to
l3kernel
\l__box_top_dim=\dimen173
\l__box_bottom_dim=\dimen174
\l__box_left_dim=\dimen175
\l__box_right_dim=\dimen176
\l__box_top_new_dim=\dimen177
\l__box_bottom_new_dim=\dimen178
\l__box_left_new_dim=\dimen179
\l__box_right_new_dim=\dimen180
\l__box_internal_box=\box53
\l__coffin_bounding_shift_dim=\dimen181
\l__coffin_left_corner_dim=\dimen182
\l__coffin_right_corner_dim=\dimen183
\l__coffin_bottom_corner_dim=\dimen184
\l__coffin_top_corner_dim=\dimen185
\l__coffin_scaled_total_height_dim=\dimen186
\l__coffin_scaled_width_dim=\dimen187
L3 Module: l3luatex 2016/03/26 v6465 L3 Experimental LuaTeX-specific
functions
) (c:/TeXLive/2015/texmf-dist/tex/latex/l3kernel/l3pdfmode.def
File: l3pdfmode.def 2016/03/26 v6465 L3 Experimental driver: PDF mode
\l__driver_color_stack_int=\count163
))
Package: breqn 2015/08/11 v0.98d Breaking equations
(c:/TeXLive/2015/texmf-dist/tex/latex/breqn/flexisym.sty
Package: flexisym 2015/08/11 v0.98d Make math characters macros
LaTeX Info: Redefining \textprime on input line 299.
LaTeX Info: Redefining \not on input line 357.
(c:/TeXLive/2015/texmf-dist/tex/latex/breqn/cmbase.sym
```

```
File: cmbase.sym 2007/12/19 v0.92
LaTeX Info: Redefining \hbar on input line 324.
LaTeX Info: Redefining \surd on input line 326.
LaTeX Info: Redefining \angle on input line 334.
LaTeX Info: Redefining \neq on input line 335.
LaTeX Info: Redefining \mapsto on input line 336.
LaTeX Info: Redefining \cong on input line 337.
LaTeX Info: Redefining \notin on input line 340.
LaTeX Info: Redefining \rightleftharpoons on input line 341.
LaTeX Info: Redefining \doteq on input line 342.
LaTeX Info: Redefining \hookrightarrow on input line 343.
LaTeX Info: Redefining \hookleftarrow on input line 344.
LaTeX Info: Redefining \bowtie on input line 345.
LaTeX Info: Redefining \models on input line 346.
LaTeX Info: Redefining \Longrightarrow on input line 347.
LaTeX Info: Redefining \longrightarrow on input line 348.
LaTeX Info: Redefining \Longleftarrow on input line 349.
LaTeX Info: Redefining \longleftarrow on input line 350.
LaTeX Info: Redefining \longmapsto on input line 351.
LaTeX Info: Redefining \longleftrightarrow on input line 352.
LaTeX Info: Redefining \Longleftrightarrow on input line 353.
LaTeX Info: Redefining \cdots on input line 357.
LaTeX Info: Redefining \vdots on input line 360.
LaTeX Info: Redefining \ddots on input line 365.
) (c:/TeXLive/2015/texmf-dist/tex/latex/breqn/mathstyle.sty
Package: mathstyle 2015/08/11 v0.98d Tracking mathstyle implicitly
LaTeX Info: Redefining \displaystyle on input line 93.
LaTeX Info: Redefining \textstyle on input line 95.
LaTeX Info: Redefining \scriptstyle on input line 97.
LaTeX Info: Redefining \scriptscriptstyle on input line 99.
LaTeX Info: Redefining \genfrac on input line 145.
))
\inf@bad=\count164
\maxint=\count165
\listwidth=\dimen188
\eqnumsep=\dimen189
\eqmargin=\dimen190
\eqlinespacing=\skip68
\eqlineskip=\skip69
\eqlineskiplimit=\dimen191
\eqbinoffset=\muskip18
\eqdelimoffset=\muskip19
\eqindentstep=\dimen192
\eqstyle=\toks31
\eqbreakdepth=\count166
\eqinterlinepenalty=\count167
\intereqpenalty=\count168
\intereqskip=\skip70
\prerelpenalty=\count169
\prebinoppenalty=\count170
\Dmedmuskip=\muskip20
\Dthickmuskip=\muskip21
\eqleftskip=\skip71
\eqrightskip=\skip72
```

```
\eq@vspan=\skip73
\eq@binoffset=\muskip22
\EQ@box=\box54
\EQ@copy=\box55
\EQ@numbox=\box56
\eq@wdNum=\dimen193
\GRP@numbox=\box57
\grp@wdNum=\dimen194
\eq@lines=\count171
\eq@curline=\count172
\eq@badness=\count173
\EQ@vims=\count174
\eq@dp=\dimen195
\eq@wdL=\dimen196
\eq@wdT=\dimen197
\eq@wdMin=\dimen198
\grp@wdL=\dimen199
\grp@wdR=\dimen256
\grp@wdT=\dimen257
\eq@wdRmax=\dimen258
\eq@firstht=\dimen259
\eq@wdCond=\dimen260
\eq@indentstep=\dimen261
\eq@linewidth=\dimen262
\grp@linewidth=\dimen263
\eq@hshift=\dimen264
\eq@given@sidespace=\dimen265
\eq@final@linecount=\count175
\eq@wdR=\dimen266
\EQ@continue=\toks32
\lr@level=\count176
\GRP@queue=\toks33
\GRP@box=\box58
\GRP@wholebox=\box59
\darraycolsep=\skip74
\cur@row=\count177
\cur@col=\count178
\conditionsep=\skip75
) (c:/TeXLive/2015/texmf-dist/tex/latex/footmisc/footmisc.sty
Package: footmisc 2011/06/06 v5.5b a miscellany of footnote facilities
\FN@temptoken=\toks34
\footnotemargin=\dimen267
\c@pp@next@reset=\count179
Package footmisc Info: Declaring symbol style bringhurst on input line
855.
Package footmisc Info: Declaring symbol style chicago on input line 863.
Package footmisc Info: Declaring symbol style wiley on input line 872.
Package footmisc Info: Declaring symbol style lamport-robust on input
line 883.

Package footmisc Info: Declaring symbol style lamport* on input line 903.
Package footmisc Info: Declaring symbol style lamport*-robust on input
line 924
.
```

```
) (./gopinathargallauro2017.aux)
\openout1 = `gopinathargallauro2017.aux'.

LaTeX Font Info:    Checking defaults for OML/cmm/m/it on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for T1/cmr/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for OT1/cmr/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for OMS/cmsy/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for OMX/cmex/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for U/cmr/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
LaTeX Font Info:    Checking defaults for TS1/cmr/m/n on input line 89.
LaTeX Font Info:    ... okay on input line 89.
(c:/TeXLive/2015/texmf-dist/tex/context/base/supp-pdf.mkii
[Loading MPS to PDF converter (version 2006.09.02).]
\scratchcounter=\count180
\scratchdimen=\dimen268
\scratchbox=\box60
\nofMPsegments=\count181
\nofMParguments=\count182
\everyMPshowfont=\toks35
\MPscratchCnt=\count183
\MPscratchDim=\dimen269
\MPnumerator=\count184
\makeMPintoPDFobject=\count185
\everyMPtoPDFconversion=\toks36
) (c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/pdftexcmds.sty
Package: pdftexcmds 2011/11/29 v0.20 Utility functions of pdfTeX for
LuaTeX (HO
)
(c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/ifluatex.sty
Package: ifluatex 2010/03/01 v1.3 Provides the ifluatex switch (HO)
Package ifluatex Info: LuaTeX not detected.
) (c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/ifpdf.sty
Package: ifpdf 2011/01/30 v2.3 Provides the ifpdf switch (HO)
Package ifpdf Info: pdfTeX in PDF mode is detected.
)
Package pdftexcmds Info: LuaTeX not detected.
Package pdftexcmds Info: \pdf@primitive is available.
Package pdftexcmds Info: \pdf@ifprimitive is available.
Package pdftexcmds Info: \pdfdraftmode found.
) (c:/TeXLive/2015/texmf-dist/tex/latex/oberdiek/epstopdf-base.sty
Package: epstopdf-base 2010/02/09 v2.5 Base part for package epstopdf
(c:/TeXLive/2015/texmf-dist/tex/latex/oberdiek/grfext.sty
Package: grfext 2010/08/19 v1.1 Manage graphics extensions (HO)
(c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/kvdefinekeys.sty
Package: kvdefinekeys 2011/04/07 v1.3 Define keys (HO)
)) (c:/TeXLive/2015/texmf-dist/tex/latex/oberdiek/kvoptions.sty
Package: kvoptions 2011/06/30 v3.11 Key value format for package options
(HO)
```

```
(c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/kvsetkeys.sty
Package: kvsetkeys 2012/04/25 v1.16 Key value parser (HO)
(c:/TeXLive/2015/texmf-dist/tex/generic/oberdiek/etexcmds.sty
Package: etexcmds 2011/02/16 v1.5 Avoid name clashes with e-TeX commands
(HO)
Package etexcmds Info: Could not find \expanded.
(etexcmds)             That can mean that you are not using pdfTeX 1.50
or
(etexcmds)             that some package has redefined \expanded.
(etexcmds)             In the latter case, load this package earlier.
)))
Package grfext Info: Graphics extension search list:
(grfext)
[.png,.pdf,.jpg,.mps,.jpeg,.jbig2,.jb2,.PNG,.PDF,.JPG,.JPE
G,.JBIG2,.JB2,.eps]
(grfext)             \AppendGraphicsExtensions on input line 452.
(c:/TeXLive/2015/texmf-dist/tex/latex/latexconfig/epstopdf-sys.cfg
File: epstopdf-sys.cfg 2010/07/13 v1.3 Configuration of (r)epstopdf for
TeX Liv
e
))
LaTeX Info: Redefining \microtypecontext on input line 89.
Package microtype Info: Generating PDF output.
Package microtype Info: Character protrusion enabled (level 2).
Package microtype Info: Using default protrusion set `alltext'.
Package microtype Info: Automatic font expansion enabled (level 2),
(microtype)             stretch: 20, shrink: 20, step: 1, non-selected.
Package microtype Info: Using default expansion set `basictext'.
Package microtype Info: No adjustment of tracking.
Package microtype Info: No adjustment of interword spacing.
Package microtype Info: No adjustment of character kerning.
(c:/TeXLive/2015/texmf-dist/tex/latex/microtype/mt-cmr.cfg
File: mt-cmr.cfg 2013/05/19 v2.2 microtype config. file: Computer Modern
Roman
(RS)
)
LaTeX Font Info:    Try loading font information for U+msa on input line
126.
(c:/TeXLive/2015/texmf-dist/tex/latex/amsfonts/umsa.fd
File: umsa.fd 2013/01/14 v3.01 AMS symbols A
) (c:/TeXLive/2015/texmf-dist/tex/latex/microtype/mt-msa.cfg
File: mt-msa.cfg 2006/02/04 v1.1 microtype config. file: AMS symbols (a)
(RS)
)
LaTeX Font Info:    Try loading font information for U+msb on input line
126.
(c:/TeXLive/2015/texmf-dist/tex/latex/amsfonts/umsb.fd
File: umsb.fd 2013/01/14 v3.01 AMS symbols B
) (c:/TeXLive/2015/texmf-dist/tex/latex/microtype/mt-msb.cfg
File: mt-msb.cfg 2005/06/01 v1.0 microtype config. file: AMS symbols (b)
(RS)
)
LaTeX Font Info:    Font shape `U/lasy/b/n' in size <7> not available
```

```
(Font)              Font shape `U/lasy/m/n' tried instead on input line
126.
LaTeX Font Info:    Font shape `U/lasy/b/n' in size <5> not available
(Font)              Font shape `U/lasy/m/n' tried instead on input line
126.
LaTeX Font Info:    Calculating math sizes for size <8.5> on input line
126.

Package natbib Warning: Citation `laplante1992assistive' on page 1
undefined on
 input line 141.


Package natbib Warning: Citation `scherer1996outcomes' on page 1
undefined on i
nput line 141.


Package natbib Warning: Citation `huete2012personal' on page 1 undefined
on inp
ut line 141.


Package natbib Warning: Citation `simpson2008tooth' on page 1 undefined
on inpu
t line 143.


Package natbib Warning: Citation `nuttin2002selection' on page 1
undefined on i
nput line 143.

Package epstopdf Info: Source file: <Fig1.eps>
(epstopdf)                    date: 2017-12-01 20:37:23
(epstopdf)                    size: 3011293 bytes
(epstopdf)             Output file: <Fig1-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:37
(epstopdf)                    size: 1698620 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig1-eps-converted-
to.pdf
Fig1.eps>
(epstopdf)             \includegraphics on input line 146.
Package epstopdf Info: Output file is already uptodate.
<Fig1-eps-converted-to.pdf, id=1, 1659.19875pt x 1281.78876pt>
File: Fig1-eps-converted-to.pdf Graphic file (type pdf)
<use Fig1-eps-converted-to.pdf>
Package pdftex.def Info: Fig1-eps-converted-to.pdf used on input line
146.
(pdftex.def)             Requested size: 420.75302pt x 325.04265pt.
LaTeX Font Info:    Font shape `U/lasy/b/n' in size <8.5> not available
(Font)              Font shape `U/lasy/m/n' tried instead on input line
147.
LaTeX Font Info:    Font shape `U/lasy/b/n' in size <5.94997> not
available
```

```
(Font)              Font shape `U/lasy/m/n' tried instead on input line
147.
LaTeX Font Info:    Font shape `U/lasy/b/n' in size <4.25> not available
(Font)              Font shape `U/lasy/m/n' tried instead on input line
147.

Package natbib Warning: Citation `eftring1999technical' on page 1
undefined on
input line 150.

[1{c:/TeXLive/2015/texmf-var/fonts/map/pdftex/updmap/pdftex.map}


]

Package natbib Warning: Citation `gopinath2017mode' on page 2 undefined
on inpu
t line 156.

Package epstopdf Info: Source file: <Fig2.eps>
(epstopdf)                    date: 2017-12-01 20:37:23
(epstopdf)                    size: 125295 bytes
(epstopdf)             Output file: <Fig2-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:40
(epstopdf)                    size: 126768 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig2-eps-converted-
to.pdf
Fig2.eps>
(epstopdf)             \includegraphics on input line 164.
Package epstopdf Info: Output file is already uptodate.
<Fig2-eps-converted-to.pdf, id=17, 663.47874pt x 478.78876pt>
File: Fig2-eps-converted-to.pdf Graphic file (type pdf)
<use Fig2-eps-converted-to.pdf>
Package pdftex.def Info: Fig2-eps-converted-to.pdf used on input line
164.
(pdftex.def)             Requested size: 173.25302pt x 125.02275pt.

Package natbib Warning: Citation `philips2007adaptive' on page 2
undefined on i
nput line 180.


Package natbib Warning: Citation `demeester2008user' on page 2 undefined
on inp
ut line 180.


Package natbib Warning: Citation `gopinath2017human' on page 2 undefined
on inp
ut line 180.


Package natbib Warning: Citation `muelling2017autonomy' on page 2
undefined on
```

```
input line 180.


Package natbib Warning: Citation `tsui2011want' on page 2 undefined on
input li
ne 180.


Package natbib Warning: Citation `kim2010relationship' on page 2
undefined on i
nput line 180.


Package natbib Warning: Citation `kim2012autonomy' on page 2 undefined on
input
 line 180.


Package natbib Warning: Citation `driessen2005collaborative' on page 2
undefine
d on input line 180.


Package natbib Warning: Citation `downey2016blending' on page 2 undefined
on in
put line 180.


Package natbib Warning: Citation `storms2014blending' on page 2 undefined
on in
put line 180.


Package natbib Warning: Citation `muelling2017autonomy' on page 2
undefined on
input line 180.

[2]
Underfull \vbox (badness 10000) has occurred while \output is active []


Package natbib Warning: Citation `herlant2016assistive' on page 3
undefined on
input line 182.


Package natbib Warning: Citation `pilarski2012dynamic' on page 3
undefined on i
nput line 182.


Package natbib Warning: Citation `liu2016goal' on page 3 undefined on
input lin
e 184.
```

```
Package natbib Warning: Citation `choi2008laser' on page 3 undefined on
input l
ine 184.


Package natbib Warning: Citation `baker2007goal' on page 3 undefined on
input l
ine 184.


Package natbib Warning: Citation `baker2009action' on page 3 undefined on
input
 line 184.


Package natbib Warning: Citation `taha2011pomdp' on page 3 undefined on
input l
ine 184.


Package natbib Warning: Citation `dragan2012assistive' on page 3
undefined on i
nput line 184.


Package natbib Warning: Citation `gopinath2017human' on page 3 undefined
on inp
ut line 184.


Underfull \vbox (badness 2828) has occurred while \output is active []

[3 <./Fig1-eps-converted-to.pdf> <./Fig2-eps-converted-to.pdf>]

Package natbib Warning: Citation `sadigh2016information' on page 4
undefined on
 input line 187.


Package natbib Warning: Citation `sadigh2016planning' on page 4 undefined
on in
put line 187.


Package natbib Warning: Citation `atanasov2014information' on page 4
undefined
on input line 187.


Package natbib Warning: Citation `miller2016ergodic' on page 4 undefined
on inp
ut line 188.
```

```
Package natbib Warning: Citation `miller2013trajectory' on page 4
undefined on
input line 188.


Package natbib Warning: Citation `tomasello2007shared' on page 4
undefined on i
nput line 190.


Package natbib Warning: Citation `tomasello2010gap' on page 4 undefined
on inpu
t line 190.


Package natbib Warning: Citation `sorokin2010people' on page 4 undefined
on inp
ut line 190.


Package natbib Warning: Citation `goodfellow2010help' on page 4 undefined
on in
put line 190.


Package natbib Warning: Citation `dragan2013legibility' on page 4
undefined on
input line 190.


Package natbib Warning: Citation `holladay2014legible' on page 4
undefined on i
nput line 190.


Package natbib Warning: Citation `gopinath2017mode' on page 4 undefined
on inpu
t line 190.


Underfull \vbox (badness 2426) has occurred while \output is active []

[4]

Package natbib Warning: Citation `kelley2008understanding' on page 5
undefined
on input line 208.


Package natbib Warning: Citation `wang2013probabilistic' on page 5
undefined on
 input line 208.
```

```
Package epstopdf Info: Source file: <Fig3.eps>
(epstopdf)                    date: 2017-12-01 20:37:23
(epstopdf)                    size: 49232 bytes
(epstopdf)             Output file: <Fig3-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:42
(epstopdf)                    size: 39004 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig3-eps-converted-
to.pdf
Fig3.eps>
(epstopdf)             \includegraphics on input line 212.
Package epstopdf Info: Output file is already uptodate.
<Fig3-eps-converted-to.pdf, id=40, 478.78876pt x 438.63875pt>
File: Fig3-eps-converted-to.pdf Graphic file (type pdf)
<use Fig3-eps-converted-to.pdf>
Package pdftex.def Info: Fig3-eps-converted-to.pdf used on input line
212.
(pdftex.def)             Requested size: 238.96417pt x 218.92393pt.
[5 <./Fig3-eps-converted-to.pdf>]
Package epstopdf Info: Source file: <Fig4.eps>
(epstopdf)                    date: 2017-12-01 20:37:23
(epstopdf)                    size: 411428 bytes
(epstopdf)             Output file: <Fig4-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:45
(epstopdf)                    size: 798504 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig4-eps-converted-
to.pdf
Fig4.eps>
(epstopdf)             \includegraphics on input line 275.
Package epstopdf Info: Output file is already uptodate.
<Fig4-eps-converted-to.pdf, id=55, 1314.9125pt x 736.7525pt>
File: Fig4-eps-converted-to.pdf Graphic file (type pdf)
<use Fig4-eps-converted-to.pdf>
Package pdftex.def Info: Fig4-eps-converted-to.pdf used on input line
275.
(pdftex.def)             Requested size: 569.25363pt x 299.59726pt.

Package natbib Warning: Citation `gopinath2017mode' on page 6 undefined
on inpu
t line 276.


Package natbib Warning: Citation `gopinath2017mode' on page 6 undefined
on inpu
t line 276.


Package natbib Warning: Citation `gopinath2017mode' on page 6 undefined
on inpu
t line 311.

[6]
Underfull \vbox (badness 10000) has occurred while \output is active []
```

```
Package natbib Warning: Citation `schoner2015dynamic' on page 7 undefined
on in
put line 320.


Package natbib Warning: Citation `amari1977dynamics' on page 7 undefined
on inp
ut line 322.

[7 <./Fig4-eps-converted-to.pdf>]

Package natbib Warning: Citation `wilson1973mathematical' on page 8
undefined o
n input line 327.


Package natbib Warning: Citation `schoner2008dynamical' on page 8
undefined on
input line 329.


Package natbib Warning: Citation `erlhagen2006dynamic' on page 8
undefined on i
nput line 330.


Package natbib Warning: Citation `erlhagen2014dynamic' on page 8
undefined on i
nput line 330.


Package natbib Warning: Citation `zibner2011dynamic' on page 8 undefined
on inp
ut line 330.


Package natbib Warning: Citation `schoner1995dynamics' on page 8
undefined on i
nput line 330.


Package natbib Warning: Citation `faubel2008learning' on page 8 undefined
on in
put line 330.

LaTeX Font Info:    Try loading font information for U+bbm on input line
344.
(c:/TeXLive/2015/texmf-dist/tex/latex/bbm-macros/ubbm.fd
File: ubbm.fd 1999/03/15 V 1.2 Font definition for bbm font - TH
) [8]

Package natbib Warning: Citation `argall2009survey' on page 9 undefined
on inpu
```

```
t line 366.


Package natbib Warning: Citation `schaal1997learning' on page 9 undefined
on in
put line 366.


Package natbib Warning: Citation `hsu2002randomized' on page 9 undefined
on inp
ut line 366.


Package natbib Warning: Citation `ratliff2009chomp' on page 9 undefined
on inpu
t line 366.


Package natbib Warning: Citation `rimon1992exact' on page 9 undefined on
input
line 366.


Package natbib Warning: Citation `tanner2003nonholonomic' on page 9
undefined o
n input line 366.


LaTeX Font Warning: Font shape `U/bbm/m/n' in size <8.5> not available
(Font)              size <8> substituted on input line 381.


LaTeX Font Warning: Font shape `U/bbm/m/n' in size <4.25> not available
(Font)              size <5> substituted on input line 381.


Package natbib Warning: Citation `khatib1986real' on page 9 undefined on
input
line 383.

Package epstopdf Info: Source file: <Fig5.eps>
(epstopdf)                    date: 2017-12-01 20:37:24
(epstopdf)                    size: 310438 bytes
(epstopdf)             Output file: <Fig5-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:48
(epstopdf)                    size: 83448 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig5-eps-converted-
to.pdf
Fig5.eps>
(epstopdf)             \includegraphics on input line 398.
Package epstopdf Info: Output file is already uptodate.
<Fig5-eps-converted-to.pdf, id=74, 1144.275pt x 418.56375pt>
File: Fig5-eps-converted-to.pdf Graphic file (type pdf)
<use Fig5-eps-converted-to.pdf>
```

```
Package pdftex.def Info: Fig5-eps-converted-to.pdf used on input line
398.
(pdftex.def)             Requested size: 238.9604pt x 93.20837pt.
Package epstopdf Info: Source file: <Fig6.eps>
(epstopdf)                     date: 2017-12-01 20:37:24
(epstopdf)                     size: 6490402 bytes
(epstopdf)              Output file: <Fig6-eps-converted-to.pdf>
(epstopdf)                     date: 2017-12-01 20:37:50
(epstopdf)                     size: 337764 bytes
(epstopdf)              Command: <repstopdf --outfile=Fig6-eps-converted-
to.pdf
Fig6.eps>
(epstopdf)              \includegraphics on input line 404.
Package epstopdf Info: Output file is already uptodate.
<Fig6-eps-converted-to.pdf, id=75, 1927.2pt x 542.025pt>
File: Fig6-eps-converted-to.pdf Graphic file (type pdf)
<use Fig6-eps-converted-to.pdf>
Package pdftex.def Info: Fig6-eps-converted-to.pdf used on input line
404.
(pdftex.def)             Requested size: 495.0pt x 139.21115pt.
[9 <./Fig5-eps-converted-to.pdf>]
Package epstopdf Info: Source file: <Fig7.eps>
(epstopdf)                     date: 2017-12-01 20:37:24
(epstopdf)                     size: 34226 bytes
(epstopdf)              Output file: <Fig7-eps-converted-to.pdf>
(epstopdf)                     date: 2017-12-01 20:37:53
(epstopdf)                     size: 10330 bytes
(epstopdf)              Command: <repstopdf --outfile=Fig7-eps-converted-
to.pdf
Fig7.eps>
(epstopdf)              \includegraphics on input line 437.
Package epstopdf Info: Output file is already uptodate.
<Fig7-eps-converted-to.pdf, id=82, 682.55pt x 371.3875pt>
File: Fig7-eps-converted-to.pdf Graphic file (type pdf)
<use Fig7-eps-converted-to.pdf>
Package pdftex.def Info: Fig7-eps-converted-to.pdf used on input line
437.
(pdftex.def)             Requested size: 450.45181pt x 245.0995pt.

Overfull \hbox (1.40538pt too wide) in paragraph at lines 447--448
$[]$\OT1/cmr/m/n/10 (-20) : In the sec-ond phase of train-ing, the
blending-
 []

[10 <./Fig6-eps-converted-to.pdf>]
Package epstopdf Info: Source file: <Fig8.eps>
(epstopdf)                     date: 2017-12-01 20:37:24
(epstopdf)                     size: 193852 bytes
(epstopdf)              Output file: <Fig8-eps-converted-to.pdf>
(epstopdf)                     date: 2017-12-01 20:37:55
(epstopdf)                     size: 215176 bytes
(epstopdf)              Command: <repstopdf --outfile=Fig8-eps-converted-
to.pdf
Fig8.eps>
```

```
(epstopdf)             \includegraphics on input line 470.
Package epstopdf Info: Output file is already uptodate.
<Fig8-eps-converted-to.pdf, id=92, 716.6775pt x 541.02126pt>
File: Fig8-eps-converted-to.pdf Graphic file (type pdf)
<use Fig8-eps-converted-to.pdf>
Package pdftex.def Info: Fig8-eps-converted-to.pdf used on input line
470.
(pdftex.def)             Requested size: 415.79819pt x 313.8916pt.
[11 <./Fig7-eps-converted-to.pdf>]
Package epstopdf Info: Source file: <Fig9.eps>
(epstopdf)                    date: 2017-12-01 20:37:24
(epstopdf)                    size: 2099967 bytes
(epstopdf)             Output file: <Fig9-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:37:58
(epstopdf)                    size: 157574 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig9-eps-converted-
to.pdf
Fig9.eps>
(epstopdf)             \includegraphics on input line 503.
Package epstopdf Info: Output file is already uptodate.
<Fig9-eps-converted-to.pdf, id=104, 619.31375pt x 257.96375pt>
File: Fig9-eps-converted-to.pdf Graphic file (type pdf)
<use Fig9-eps-converted-to.pdf>
Package pdftex.def Info: Fig9-eps-converted-to.pdf used on input line
503.
(pdftex.def)             Requested size: 396.00151pt x 164.95451pt.
[12 <./Fig8-eps-converted-to.pdf>]
Package epstopdf Info: Source file: <Fig10.eps>
(epstopdf)                    date: 2017-12-01 20:37:24
(epstopdf)                    size: 28366 bytes
(epstopdf)             Output file: <Fig10-eps-converted-to.pdf>
(epstopdf)                    date: 2017-12-01 20:38:00
(epstopdf)                    size: 8842 bytes
(epstopdf)             Command: <repstopdf --outfile=Fig10-eps-converted-
to.pdf
 Fig10.eps>
(epstopdf)             \includegraphics on input line 515.
Package epstopdf Info: Output file is already uptodate.
<Fig10-eps-converted-to.pdf, id=114, 619.31375pt x 447.6725pt>
File: Fig10-eps-converted-to.pdf Graphic file (type pdf)
<use Fig10-eps-converted-to.pdf>
Package pdftex.def Info: Fig10-eps-converted-to.pdf used on input line
515.
(pdftex.def)             Requested size: 238.96417pt x 172.73355pt.
[13 <./Fig9-eps-converted-to.pdf> <./Fig10-eps-converted-to.pdf>]
No file gopinathargallauro2017.bbl.

Package natbib Warning: There were undefined citations.


Package balance Warning: You have called \balance in second column
(balance)                Columns might not be balanced.

[14] (./gopinathargallauro2017.aux)
```

```
LaTeX Font Warning: Size substitutions with differences
(Font)              up to 0.75pt have occurred.

 )
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \iffalse on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \iffalse on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \iffalse on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \iffalse on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 89 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \iffalse on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \iffalse on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \iffalse on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \iffalse on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 86 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \iffalse on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \iffalse on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \iffalse on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \iffalse on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \ifx on line 81 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
```

```
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \iffalse on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \ifx on line 71 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \iffalse on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
(\end occurred when \ifx on line 64 was incomplete)
Here is how much of TeX's memory you used:
 14852 strings out of 493027
 253889 string characters out of 6137679
 424594 words of memory out of 5000000
 17977 multiletter control sequences out of 15000+600000
 24687 words of font info for 168 fonts, out of 8000000 for 9000
 1141 hyphenation exceptions out of 8191
```

```
 38i,20n,51p,10404b,590s stack positions out of
5000i,500n,10000p,200000b,80000s
 <c:/Users/asynch/.texlive20
15/texmf-
var/fonts/pk/ljfour/public/bbm/bbm10.600pk><c:/TeXLive/2015/texmf-dist
/fonts/type1/public/amsfonts/cm/cmbsy10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/t
ype1/public/amsfonts/cm/cmbx10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/publ
ic/amsfonts/cm/cmbx12.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfon
ts/cm/cmbx8.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmex
10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmmi10.pfb><c
:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmmi5.pfb><c:/TeXLive/
2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmmi7.pfb><c:/TeXLive/2015/texmf
-dist/fonts/type1/public/amsfonts/cm/cmmi8.pfb><c:/TeXLive/2015/texmf-
dist/font
s/type1/public/amsfonts/cm/cmmib10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/
public/amsfonts/cmextra/cmmib7.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/publ
ic/amsfonts/cm/cmr10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfont
s/cm/cmr5.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmr7.p
fb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmr8.pfb><c:/TeXL
ive/2015/texmf-
dist/fonts/type1/public/amsfonts/cm/cmsy10.pfb><c:/TeXLive/2015/
texmf-
dist/fonts/type1/public/amsfonts/cm/cmsy5.pfb><c:/TeXLive/2015/texmf-dist
/fonts/type1/public/amsfonts/cm/cmsy7.pfb><c:/TeXLive/2015/texmf-
dist/fonts/typ
e1/public/amsfonts/cm/cmsy8.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/
amsfonts/cm/cmti10.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/
cm/cmti8.pfb><c:/TeXLive/2015/texmf-
dist/fonts/type1/public/amsfonts/symbols/ms
bm10.pfb>
Output written on gopinathargallauro2017.pdf (14 pages, 3779337 bytes).
PDF statistics:
 206 PDF objects out of 1000 (max. 8388607)
 131 compressed objects within 2 object streams
 0 named destinations out of 1000 (max. 500000)
 14387 words of extra memory for PDF output out of 14400 (max. 10000000)
```

**Autonomous Robots manuscript No.**
(will be inserted by the editor)

# Disambiguation of Human Intent Through Control Space Selection

**Deepak E. Gopinath*** · **Brenna D. Argall**

Received: date / Accepted: date

**Abstract** Assistive shared-control robots have the potential to transform the lives of millions of people afflicted with severe motor impairments as a result of spinal cord or brain injuries. The effectiveness and usefulness of shared-control robots is closely related to their ability to infer the user's needs and intentions and is often a limiting factor for providing appropriate assistance quickly, confidently and accurately. The contributions of this paper are three-fold: first, we propose a goal disambiguation algorithm which enhances the intent inference and assistive capabilities of a shared-control assistive robotic arm. Second, we introduce a novel intent inference algorithm that works in conjunction with the disambiguation scheme, inspired by *dynamic field theory* in which the time evolution of the probability distribution over goals is specified as a dynamical system. Third, we present a pilot human subject study to evaluate the efficacy of the disambiguation system. This study was performed with eight subjects. Our results suggest that (a) the disambiguation system has a greater utility value as the control interface becomes more limited and the task becomes more complex, (b) subjects demonstrated a diverse range of disambiguation request behavior with a greater concentration in the earlier parts of the trial and (c) there are no differences in the onset of robot assistance between different mode switching paradigms across tasks or across interfaces.

**Keywords** Shared Autonomy · Intent Inference · Intent Disambiguation · Assistive Robotics

Deepak E. Gopinath
Department of Mechanical Engineering
Northwestern University, Evanston, IL
Shirley Ryan AbilityLab, Chicago, IL.
Phone No: +12013109061
E-mail: deepakgopinath@u.northwestern.edu

Brenna D. Argall
Department of Mechanical Engineering, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL.
Department of Physical Medicine and Rehabilitation, Northwestern University, Chicago, IL,
Shirley Ryan AbilityLab, Chicago, IL.
E-mail: brenna.argall@northwestern.edu

* Corresponding Author

**Acknowledgements** This material is based upon work supported by the National Science Foundation under Grant CNS 1544741. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the aforementioned institutions.

## 1 Introduction

Assistive and rehabilitation machines—such as robotic arms and smart wheelchairs—have the potential to transform the lives of millions of people with severe motor impairments (?). These machines can promote independence, boost self-esteem and help to extend the mobility and manipulation capabilities of such individuals, and revolutionize the way motor-impaired people interact with society (??). With rapid technological strides in the domain of assistive robotics, the machines have become more capable and complex, and with this complexity the control of these machines has become a greater challenge.

The control of an assistive machine is typically enacted through a control interface. Moreover, the greater the motor impairment of the user, the more limited are the interfaces available for them to use. These interfaces (for example, sip-and-puffs and switch-based head arrays) are low-dimensional, discrete interfaces that can operate only in subsets of the entire control space (??).

The dimensionality mismatch between the control interfaces and the controllable degrees-of-freedom of the assistive robot necessitates the partitioning of the entire control space into smaller subsets called *control modes*. Moreover, the more limited and lower-dimensional the control interface, the greater the number of control modes. In order to achieve full control of the machine, the user must switch between the control modes, which is referred to as *mode switching* or *modal control*. Mode switching adds to the cognitive and physical burden of task execution and has a detrimental effect on the performance (?). The introduction of *shared autonomy* to these assistive machines seeks to alleviate some of these issues. In a shared control system the task responsibility is shared between the user and the robot, with the aim of reducing human effort in achieving a goal. Shared autonomy systems arbitrate between the human control commands and the autonomous control commands using different strategies depending on the task context, user preference and robotic platform. Figure 1 depicts the key components of a typical shared control architecture.

Any assistive robotic system needs to have a good idea of the user's needs and intentions. Therefore, intent inference is a necessary and crucial component to ensure appropriate assistance. In assistive robotic manipulation specifically, often the first step of a manipulation task is to reach for and grasp discrete objects in the environment. Intent inference therefore can be framed as a problem of estimating a probability distribution of intent likelihood over all possible goals (objects) in the environment. This inference is usually informed by various cues from the human and the environment, such as the human control actions, biometric measures that indicate the cognitive and physical load of the user during task execution and task-relevant features such as robot and goal locations. With a greater number of sensor modalities available, it is likely that the intent inference becomes more accurate.

However, in the assistive domain, user acceptance and adoption is of paramount importance. Adding more sensors to track biometric data and object locations can become expensive or impractical (e.g. if the sensor must be worn by the user). For reasons of user adoption and cost, we intentionally design our assistance add-ons to be as invisible and close to the manual system as possible. The information we are able to capture from the human therefore is largely restricted to the control commands they issue to the assistive machine. Sparsity and noise in these control commands make the inference task even harder, prompting the need for robust intent inference formalisms.

Our key insight is that certain control commands issued by the human are *more intent expressive* than others, and may help the autonomy in inference accuracy. This is the notion of *inverse legibility* (?) in which human-generated actions *help the robot* to infer the human's intent unambiguously. Consider the hypothetical reaching experiment illustrated in Figure 2. Since the spatial locations of the goals are maximally spread along the horizontal axis, any human control command issued along the horizontal dimension conveys a lot of information about the intended goal to the robot. In other words, motion along $x$ is more *intent expressive* and will help the robot to draw accurate inference more quickly and confidently. This approach to more seamless human-robot interaction exploits the underlying implicit exchanges of information between partners that are inherent to task execution with shared intentions.

In this work, as our primary contribution we develop a mode switch assistance paradigm that enhances the robot's intent inference capabilities, by selecting the control mode in which a user-initiated motion will *maximally disambiguate* human intent. The disambiguation layer elicits more *intent expressive* commands from the user by placing the user control in certain control modes. Furthermore, the disambiguation power of the algorithm is closely linked to, and is dependent on, the success and accuracy of the underlying intent inference mechanism. Therefore, as our secondary contribution, we also develop a novel intent inference scheme which utilizes ideas from *dynamic field theory* that efficiently incorporates information contained in past states to improve the performance of the disambiguation system.

In Section 2 we present an overview of relevant research in the areas of shared autonomy in assistive robotics, types of shared autonomy assistance paradigms, intent inference and synergies in human-robot interaction. Section 3 presents the mathematical formalism developed for intent inference and disambiguation. Section 4 focuses on the implementation details of the shared control system. The study design and experimental methods are discussed in Section 5 followed by results in Section 6. Discussion and conclusions are presented in Sections 7 and 8 respectively.

## 2 Related Work

This section provides an overview of related research in the domains of shared autonomy in assistive robotics, robot assistance for modal control, intent inference in human-robot interaction and information acquisition in robotics.

Shared-autonomy in assistive systems aims to reduce the user's cognitive and physical burden during task execution without having the user relinquish complete control (????). The most common strategies to share

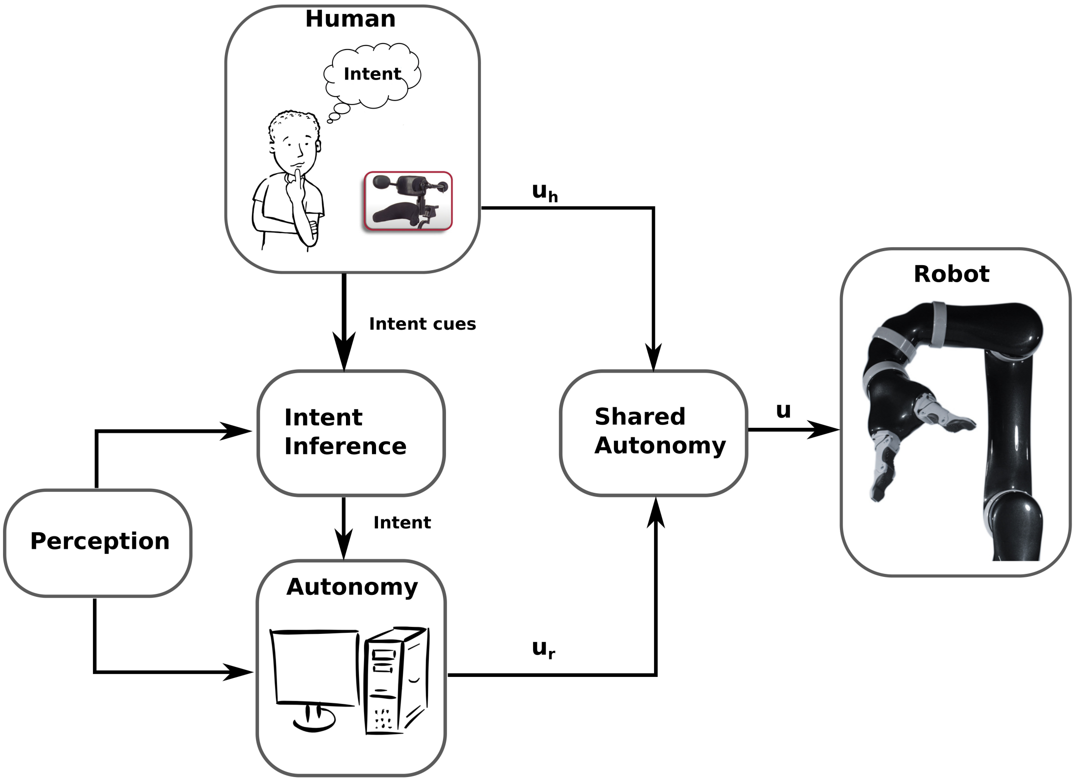


**Fig. 1** Illustration of the core components of a shared control architecture. $\boldsymbol{u}_h$ denotes the human control command, $\boldsymbol{u}_r$ denotes the autonomy control command and $\boldsymbol{u}$ is the final control command issued to the robot. The intent inference module can extract intent cues from the human in multiple ways, for example via the human control actions or biometric measures that indicate the cognitive and physical load of the user during task execution. In our work we intentionally restrict this to be similar to operating the machine without autonomy and so intent information is $\boldsymbol{u}_h$ exclusively.

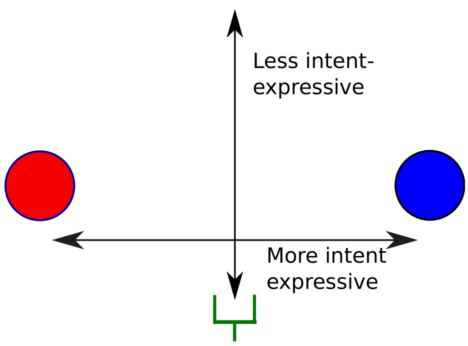


**Fig. 2** Illustration of goal disambiguation along various control dimensions. Any motion of the end effector (green) along the y-axis will not help the system to disambiguate the two goals (A and B). However, motion along the x-axis provides cues as to which goal.

control between the user and the assistive system include (a) hierarchical paradigms in which the higher level goals are entrusted with the user and the autonomy generates low-level control (???), (b) control allocation in distinct partitions of the control space (?) and (c) blending user controls and robot autonomy commands (???).

In order to offset the drop in task performance due to shifting focus (task switching) from the task at hand to switching between different control modes different mode switch assistance paradigms have been proposed. For example, a simple time-optimal mode switching scheme has shown to improve task performance (??).

Shared control systems often require a good estimate of the human's intent—for example, their intended reaching target in a manipulation task or a target location in the environment in a navigation task (?). Intent can either be explicitly communicated by the user (?) or can be inferred from their control signals or sensor data using various algorithms. Intent recognition and inference are actively studied by cognitive scientists and roboticists and can be broadly categorized into two main classes: model-based approaches and heuristic approaches. In model-based approaches, intent inference is typically cast within a Bayesian framework, and the posterior distribution over goals (belief) at any time is determined by the iterative application of Bayes theorem. Evidence in this context can be derived from a combination of factors such as task-relevant features in the environment, human control actions or biometric data from the user (??). The user often is modeled as a Partially Observable Markov Decision Process (POMDP) (?) and is assumed to behave according to a predefined control policy that maps the states to actions. Although iterative belief updating using Bayes theorem provides an optimal strategy to combine new evidence (likelihood) with *a priori* information (prior), incorporating an extended history of past states and control actions increases the computational complexity and tractability becomes an issue. In such cases, first-order Markovian independence assumptions make the inference tractable. In heuristic approaches, the formulations are often simpler and seek to find direct mappings from instantaneous cues and the underlying human intention. For example, the use of instantaneous confidence functions for estimating intended reaching target in robotic manipulation (??). However, heuristic methods in general are not sophisticated enough to incorporate histories of past states and actions, making them less robust to external noise.

Eliciting more legible and information-rich control commands from the user to improve intent estimation can be thought of as an information acquisition process. Intent information acquisition can be an *active* process in which the robot takes actions that will probe the human's intent (??). Designing optimal control laws that maximize information gain can be accomplished by having the associated reward structure reflect some measure of information gain (?). Autonomous robots designed for exploration and data acquisition tasks can benefit from exploring more information-rich regions in the environment. If the spatial distribution of information density is known *a priori*, information maximization can be accomplished by maximizing the ergodicity of the robot's trajectory with respect to the underlying information density map (??).

By having the human assist the robot in its intent inference capacity, our work leverages the underlying synergies that are inherent in human-robot cooperation. In the context of human-human cooperative teams, the notion of shared intentionality—one in which all parties involved in a collaborative task team share the same intention or goal and have a joint commitment towards it—is crucial to make task execution more seamless and efficient (??). This principle is relevant to successful human-robot interaction as well. From the robot's perspective, the core idea behind our disambiguation system is that of *"Help Me, Help You"*—that is, if the user can help the robot with more information-rich actions, then the robot in turn can provide accurate and appropriate task assistance more quickly and confidently. A framework for *"people helping robots helping people"* in which the robot relies on semantic information and judgments provided by the human to improve its own capabilities has been developed in (?). In order to overcome the various types of communication bottlenecks that can hamper performance, different types of communication interfaces have been developed that account for the restricted capabilities of the robot (?). Lastly, more intent-expressive human actions is related to the idea of legibility in robot actions. In human-robot interaction, the legibility and predictability of robot motion *to* the human has been investigated (?) and various techniques to generate legible robot motion have been proposed (?). Our work introduces the idea of *inverse legibility* (?) in which the assistance scheme is intended to bring out more legible intent-expressive control commands *from* the human.

## 3 Mathematical Formalism

This section describes our intent disambiguation algorithm that computes the control mode that can maximally disambiguate between the goals and the intent inference mechanism that works in conjunction with the disambiguation algorithm. Section 3.1 outlines the mathematical notation used in this paper. Section 3.2 describes the disambiguation algorithm. The mathematical details of the intent inference paradigms is outlined in detail in Section 3.3.

### 3.1 Notation

Let $\mathcal{G}$ denote the set of all candidate goals with $n_g = |\mathcal{G}|$ and let $g^i$ refer to the $i^{th}$ goal with $i \in [1, 2, \ldots, n_g]$. A *goal* represents the human's underlying intent. Specifically, in assistive robotic manipulation, since the robotic device is primarily used for reaching toward and grasping of discrete objects in the environment, intent inference is the estimation of the probability distribution over all possible goals (objects) in the environment. At any time $t$, the robot actively maintains a probability distribution over goals denoted by $\boldsymbol{p}(t)$ such that $\boldsymbol{p}(t) = [p^1(t), p^2(t), \ldots, p^{n_g}(t)]^T$ where $p^i(t)$ denotes the probability associated with goal $g^i$. The probability $p^i(t)$ represent the robot's *confidence* that goal $g^i$ is the human's intended goal.

Let $\mathcal{K}$ be the set of all controllable dimensions of the robot and $k^i$ represent the $i^{th}$ control dimension where $i \in [1, 2, \ldots, n_k]$. The cardinality of $\mathcal{K}$ is denoted as $n_k$ and typically depends on the robotic platform used. For example, for a smart wheelchair $n_k = 2$, since the controllable degrees-of-freedom are velocity and heading and for a six degrees-of-freedom robotic arm with a gripper $n_k = 7$.

The limitations of the control interfaces necessitate the control space $\mathcal{K}$ to be partitioned into control modes. Let $\mathcal{M}$ denote the set of all control modes with $n_m = |\mathcal{M}|$. Additionally, let $m^i$ refer to the $i^{th}$ control mode where $i \in [1, 2, \ldots, n_m]$. Each control mode $m^i$ is a subset of $\mathcal{K}$ such that $\bigcup_{i=1}^{n_m} m^i$ spans all of the controllable dimensions.[1] Let $\boldsymbol{e}^i$ be the standard basis vectors that denote the unit velocity vector along the $i^{th}$ control dimension.[2] The disambiguation formalism developed in Section 3.2 is agnostic to the particular form of intent inference. However, the algorithm assumes that $\boldsymbol{p}(t)$ can be forward projected in time by iteratively applying the intent inference algorithm.

The disambiguation metric that characterizes the disambiguation capabilities of a control dimension $k \in \mathcal{K}$ is denoted by $D_k \in \mathbb{R}$. We explicitly define disambiguation metrics for both positive negative motions along $k$ as $D_k^+$ and $D_k^-$ respectively. We also define a disambiguation metric $D_m \in \mathbb{R}$ for each control mode $m \in \mathcal{M}$. $D_m$ is a measure of how informative and useful the user control commands would be for the robot if the user were to operate the robot in control mode $m$. The higher it is, the easier it will be for the system to infer human's intent. Both $D_k$ and $D_m$ will be formally defined in Section 3.2.2.

The robot pose and the goal pose for $g \in \mathcal{G}$ are denoted $\boldsymbol{x}_r$ and $\boldsymbol{x}_g$ respectively and $\boldsymbol{u}_h$ denotes the human control command.

## 3.2 Intent Disambiguation

The need for intent disambiguation arises from how the probability distribution over goals evolves as the user controls the robot and moves it in space. That is, given an intent inference mechanism that is dependent on robot pose or movement (**??**), as the user controls the robot in different control modes, the probability distribution evolves.[3] Figure 3 shows simulations which motivate the development of a disambiguation metric. For different control modes, the confidences associated with each goal are different. Moreover, motions in some control modes result in sharper rise in some goal confidences compared to others. This indicates the existence of control modes that can better disambiguate between the goals.

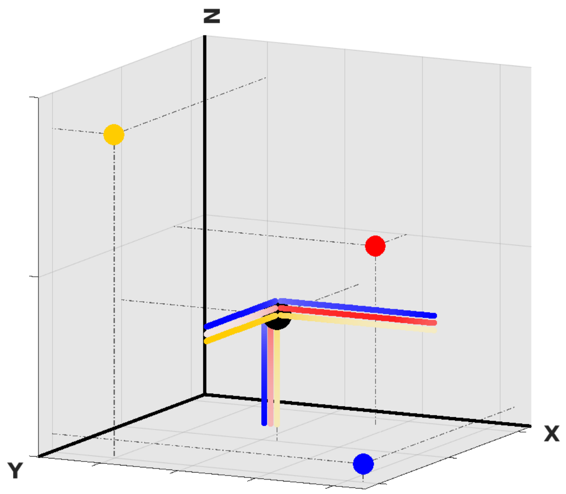


**Fig. 3** Illustration of how goal confidences change upon motion along control dimensions. The three colored lines along each dimension represent the confidences associated with a goal of corresponding color. Brighter and dimmer colors correspond to high and low confidences respectively. One can see that motion along certain control dimensions results in sharper rise in confidences for some goals compared to others.

The computation of $D_k$ depends on four components (denoted as $\Gamma_k$, $\Omega_k$, $\Lambda_k$ and $\Upsilon_k$), which in turn depend on a projection of the probability distribution over intent. These projection and computations are described in detail in Section 3.2.1 and Section 3.2.2, and as a pseudocode in Algorithm 1.

### *3.2.1 Forward Projection of $\boldsymbol{p}(t)$*

The first step towards the computation of $D_k$ is the forward projection of the probability distribution $\boldsymbol{p}(t)$ from the current time $t_a$ to $t_b$ and $t_c$ ($t_a < t_b < t_c$), Algorithm 1, lines 3-13). We consider two projection times ($t_b$ and $t_c$) in order to compute short-term and long-term behavior and evolution of the probability distribution. Application of control command $\boldsymbol{e}^k$ results in probability distributions $\boldsymbol{p}_k^+(t_b)$, $\boldsymbol{p}_k^+(t_c)$ and $-\boldsymbol{e}^k$ results in $\boldsymbol{p}_k^-(t_b)$ and $\boldsymbol{p}_k^-(t_c)$. Note that Algorithm 1 is run twice to compute the projected probability distributions for $\boldsymbol{e}^k$ and $-\boldsymbol{e}^k$.

The exact computation of the projected probability distribution will depend on the underlying intent inference computation—for example, whether it depends on

[1] Note that a dimension $k \in \mathcal{K}$ can be an element of multiple control modes.

[2] For the rotational control dimensions, the velocity is specified with respect to the end-effector of the robotic frame.

[3] Different approaches for intent inference that do *not* depend on robot motion or pose exist as well. For example, those that depend on eye gaze, electromyographic signals and brain signals.

**Algorithm 1** Calculate $\boldsymbol{p}(t_b)$, $\boldsymbol{p}(t_c)$

```
Require: p(t_a), x_r(t_a), Δt, t_a < t_b < t_c, Θ
 1: for k = 0 ... n_k do
 2:     Initialize D_k = 0, t = t_a
 3:     while t ≤ t_c do
 4:         p_k(t + Δt) ← UpdateIntent(p_k(t), u_h; Θ)
 5:         x_r(t + Δt) ← SimulateKinematics(x_r(t), u_h)
 6:         if t = t_b then
 7:             Compute Γ_k, Ω_k, Λ_k
 8:         end if
 9:         if t = t_c then
10:             Compute Υ_k
11:         end if
12:         t ← t + Δt
13:     end while
14:     Compute D_k
15: end for
```

$\boldsymbol{x_r}$ (which can be computed from $\boldsymbol{e}^k$ applied to the robot kinematics model) or $\boldsymbol{u}_h$ (which can be taken as $\boldsymbol{e}^k$). All parameters and features which affect the computation of $\boldsymbol{p}(t)$ are denoted as $\boldsymbol{\Theta}$.

### *3.2.2 Components of $D_k$*

The computation of disambiguation metric $D_k$ consists of four components. Each of the following components encodes some aspect of the shape of the probability distribution and is computed for projections along both positive and negative directions independently. The four components are computed in lines 7 and 10 in Algorithm 1.

1) *Maximum probability:* The maximum of the projected probability distribution $\boldsymbol{p}_k(t_b)$ is a good measure of the robot's overall certainty in accurate predicting human intent (The maximum of this discrete probability distribution is the mode of the distribution). A higher value implies that the robot has a good idea of which goal is the humans's intended goal. We define the distribution maximum as $\Gamma_k$.

$$\Gamma_k = \max_{1 \le i \le n_g} p_k^i(t_b) \tag{1}$$

2) *Difference between largest probabilities:* Disambiguation accuracy benefits from greater differences between the first and second most probable goals. This difference is denoted as $\Omega_k$.

$$\Omega_k = \max(\boldsymbol{p}_k(t_b)) - \max(\boldsymbol{p}_k(t_b) \setminus \max(\boldsymbol{p}_k(t_b))) \tag{2}$$

3) *Pairwise separation of probabilities:* If the difference between the largest probabilities fails to disambiguate, then the separation, $\Lambda_k$, in the remaining goal probabilities will further aid in intent disambiguation. The quantity $\Lambda_k$ is computed as the *sum of the pairwise distances* between the $n_g$ probabilities.

$$\Lambda_k = \sum_{i=1}^{n_g} \sum_{j=i}^{n_g} |p_k^i(t_b) - p_k^j(t_b)| \tag{3}$$

4) *Gradients:* The probability distribution $\boldsymbol{p}_k(t)$ can undergo drastic changes upon continuation of motion along control dimension $k$. The spatial gradient of $\boldsymbol{p}_k(t)$ encodes this propensity for change and is approximated by

$$\frac{\partial \boldsymbol{p}_k(t)}{\partial x_k} = \boldsymbol{p}_k(t_c) - \boldsymbol{p}_k(t_b)$$

where $x_k$ is the component of robot's displacement along control dimension $k$. The greater the difference between individual spatial gradients, the greater will the probabilities deviate from each other, thereby helping in disambiguation. In order to quantify the "spread" of gradients we define a quantity $\Upsilon_k$

$$\Upsilon_k = \sum_{i=1}^{n_g} \sum_{j=i}^{n_g} \Big| \frac{\partial p_k^i(t)}{\partial x_k} - \frac{\partial p_k^j(t)}{\partial x_k} \Big| \tag{4}$$

where $|\cdot|$ denotes the absolute value.

*Putting it all together:* $\Gamma_k$, $\Omega_k$, $\Lambda_k$ and $\Upsilon_k$ are then combined to compute $D_k$ as

$$D_k = \underbrace{w \cdot (\Gamma_k \cdot \Omega_k \cdot \Lambda_k)}_{\text{short-term}} + \underbrace{(1-w) \cdot \Upsilon_k}_{\text{long-term}} \tag{5}$$

where $w$ is a task-specific weight that balances the contributions of the short-term and long-term components. (In our implementation, $w = 0.5$.) Equation 5 actually is computed twice, once in each of the positive ($\boldsymbol{e}^k$) and negative directions ($-\boldsymbol{e}^k$) along $k$, and the results ($D_k^+$ and $D_k^-$) are then summed. The computation of $D_k$ is performed for each control dimension $k \in \mathcal{K}$. The disambiguation metric $D_m$ for control mode $m$ then is calculated as

$$D_m = \sum_k D_k$$

where $k \in m$ iterates through the set of control dimensions on which $m$ is able to operate. Lastly, the control mode with highest disambiguation capability $m^*$ is given by

$$m^* = \operatorname*{argmax}_m D_m$$

while $k^* = \mathrm{argmax}_k\, D_k$ gives the control dimension with highest disambiguation capability $k^*$. Disambiguation mode $m^*$ is the mode that the algorithm chooses *for* the human to better estimate their intent. Any control command issued by the user in $m^*$ is likely to be more useful for the robot in determining which is the human's intended goal, because of the maximal confidence disambiguation.

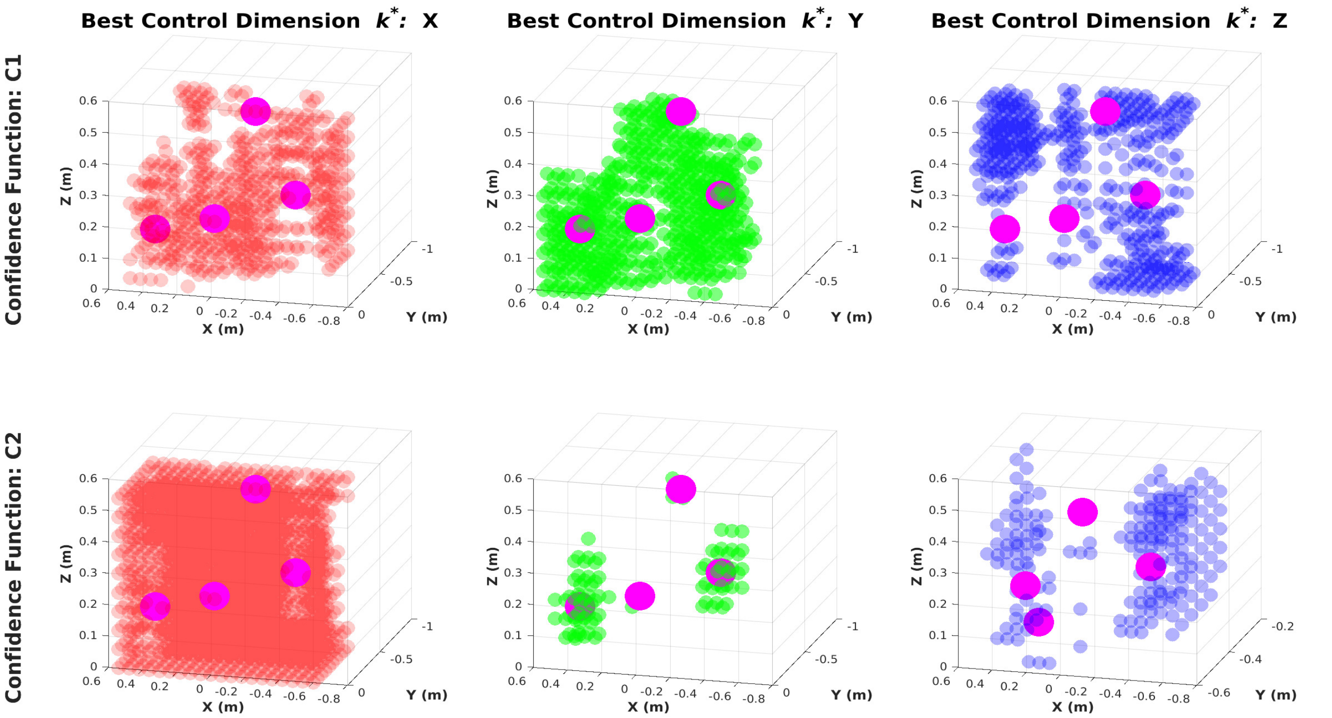


**Fig. 4** Control dimensions best able to disambiguate intent. Left column: $k^*$ is X. Middle Column: $k^*$ is Y. Right Column: $k^*$ is Z. Magenta spheres indicate the goal locations (intent). In this example, the goals are spread maximally along the $x$ and $z$ dimensions, and so inference happens more quickly if the human control commands are along $x$ or $z$. We see that $x$ and $z$ are chosen more often as the most disambiguating dimensions when using intent inference function C2 (bottom row). Function C2 considers the instantaneous directedness of the human's control command towards that goal, while inference function C1 (top row) encodes only proximity to a given goal. Function C2 is considered to encode more information about the human's intent than C1, with the result of stronger inference power—which is inherently linked to the disambiguation power of our algorithm. Further details in (?).

## 3.3 Intent Inference

This section describes the intent inference scheme used in this paper. Our preliminary work (?) revealed that the power of our disambiguation algorithm proposed in Section 3.2 is intimately linked with the inference power of different choices of intent inference mechanisms. More importantly, the pilot study associated with this preliminary work suggested that incorporating a history of past states and actions would improve performance. We therefore propose an extended disambiguation formulation which furthermore incorporates history.

In this work, we propose a novel intent inference scheme inspired by *dynamic field theory* in which the time evolution of the probability distribution $\boldsymbol{p}(t)$ is specified as a dynamical system with constraints. An alternate approach is to perform intent inference using Bayesian techniques, which in theory can take into account the influence of past states and actions. In practice, however, low-order Markov assumptions are usually made to make the inference tractable computationally, and with such assumptions history is lost.

Section 3.3.1 provides a primer on the basic principles and features of *dynamic field theory* and its application in the fields of neuroscience and cognitive robotics. Section 3.3.2 describes our novel formulation that makes use of dynamic field theory for the purposes of intent inference.

### *3.3.1 Dynamic Field Theory*

In Dynamic Field Theory (DFT) (?), variables of interest are treated as dynamical state variables. To represent the information about these variables requires two dimensions: one which specifies the value the variables can attain (the domain) and the other which encodes the *activation level* or the amount of information about that a particular value. These *activation fields* are analogous to probability distributions defined over a random variable.

Following Amari's formulation (?) dynamics of an activation field $\phi(x,t)$ are given by

$$\tau\dot{\phi}(x,t) = -\phi(x,t) + h + S(x,t) + \int dx' b(x-x')\sigma(\phi(x',t))$$

where $x$ denotes the variable of interest, $t$ is time, $\tau$ is the time-scale parameter, $h$ is the constant resting level, and $S(x,t)$ is the external input, $b(x-x')$ is the interaction kernel and $\sigma(\phi)$ is a sigmoidal nonlinear threshold function. The interaction kernel mediates how activations at all other field sites $x'$ drive the activation level at $x$. Two types of interactions are possible: excitatory (when interaction is positive) which drives up the activation, and inhibitory (when the interaction is negative) which drives the activation down. Historically, dynamic neural fields originally were conceived to explain cortical population neuronal dynamics, based on the hypothesis that the excitatory and inhibitory neural interactions between local neuronal pools form the basis of cortical information processing (?).

Dynamic neural fields possess some unique characteristics that make them ideal candidates for modeling higher-level cognition. First, a peak in the activation field can be *sustained* even in the absence of external input due to the recurrent interaction terms. Second, information from the past can be *preserved* over much larger time scales quite easily by tuning the time-scale parameter thereby endowing the fields with memory. Third, the activation fields are *robust* to disturbance and noise in the external output (?). As a result, DFT principles have found widespread application in the area of cognitive robotics (?), specifically in the contexts of efficient human-robot interaction (?), robotic scene representation (?), obstacle avoidance and target reaching behaviors in both humans and robots (?), and for object learning and recognition (?).

### 3.3.2 Dynamic Neural Fields for Intent Inference

Recurrent interaction between the state variables, robustness to noise and inherent memory make dynamic neural fields an ideal candidate for an intent inference engine. Our insight is to use the framework of dynamic neural fields to specify the time evolution of the probability distribution $\boldsymbol{p}(t)$, in which we treat the individual goal probabilities $p^i(t)$ as constrained dynamical state variables such that $p^i(t) \in [0,1]$ and $\Sigma_1^{n_g} p^i(t) = 1$. The dynamical system can be generically written as

$$\dot{\boldsymbol{p}}(t) = F(\boldsymbol{p}(t), \boldsymbol{u}_h; \boldsymbol{\Theta})$$

where $F$ represents the nonlinear vector field, $\boldsymbol{u}_h$ is the human control input and $\boldsymbol{\Theta}$ represents all other task-relevant features and parameters that affect the time-evolution of the probability distribution. The full specification of the neural field is given by

$$\frac{\partial \boldsymbol{p}(t)}{\partial t} = \frac{1}{\tau}\Bigg[ -\mathbb{I}_{n_g\times n_g}\cdot \boldsymbol{p}(t) + \underbrace{\frac{1}{n_g}\cdot \mathbb{1}_{n_g}}_{\text{rest state}}\Bigg] + \underbrace{\boldsymbol{\lambda}_{n_g\times n_g}\cdot \sigma(\boldsymbol{\xi}(\boldsymbol{u}_h;\boldsymbol{\Theta}))}_{\text{excitatory + inhibitory}} \quad (6)$$

where time-scale parameter $\tau$ which determines the memory capacity of the system, $\mathbb{I}$ is the identity matrix, $\boldsymbol{\lambda}$ is the control matrix that controls the excitatory and inhibitory aspects, $\boldsymbol{\xi}$ is a function that encodes the nonlinearity through which human control commands and task features affect the time evolution, and $\sigma$ is a biased sigmoidal nonlinearity given by $\sigma(\boldsymbol{\xi}) = \frac{1}{1+e^{-\boldsymbol{z}}} - 0.5$. The off-diagonal elements of $\boldsymbol{\lambda}$ mediate the interaction between all of the probabilities. In the absence of any information or cues, the probability distribution settles to a resting state which is a uniform distribution, that is whenever $\boldsymbol{u}_h = 0$, $\boldsymbol{\xi} = \boldsymbol{0}$. Given the initial probability distribution at time $t_a$ Equation 6 can be solved numerically from $t \in [t_a, t_b]$ using a simple Euler algorithm with a fixed time-step $\Delta t$.

The design of $\boldsymbol{\xi}$ is informed by what features of the human control input and environment capture the human's underlying intent most effectively. We rely on the *directedness* of the human control commands towards a goal, the *proximity* to a goal and the *agreement* between the human commands and robot autonomy. With $\boldsymbol{\Theta} = \{\boldsymbol{x}_r, \boldsymbol{x}_{g^i}, \boldsymbol{u}_{r,g^i}\}$, one dimension $i$ of $\boldsymbol{\xi}$ is defined as

$$\xi^i(\boldsymbol{u}_h; \boldsymbol{x}_r, \boldsymbol{x}_{g^i}, \boldsymbol{u}_{r,g^i}) = \underbrace{\frac{1+\eta}{2}}_{\text{directedness}} + \underbrace{\boldsymbol{u}_h^{rot}\cdot \boldsymbol{u}_{r,g^i}^{rot}}_{\text{agreement}} + \underbrace{\max\Big(0, 1 - \frac{\|\boldsymbol{x}_{g^i} - \boldsymbol{x}_r\|}{R}\Big)}_{\text{proximity}}$$

where $\eta = \frac{\boldsymbol{u}_h^{trans}\cdot(\boldsymbol{x}_{g^i}-\boldsymbol{x}_r)^{trans}}{\|\boldsymbol{u}_h^{trans}\|\|(\boldsymbol{x}_{g^i}-\boldsymbol{x}_r)^{trans}\|}$, $\boldsymbol{u}_{r,g^i}$ is the robot autonomy command for reaching goal $g^i$, *trans* and *rot* refer to the translational and rotational components of a command $\boldsymbol{u}$ or position $\boldsymbol{x}$, $R$ is the radius of the sphere beyond which the proximity component is always zero, and $\|\cdot\|$ is the Euclidean norm. That is, in the absence of any human control command, the probability distribution decays to the resting state which is a uniform distribution. The most confident goal $g^*$ then is computed as

$$g^* = \underset{i}{\operatorname{argmax}}\, p^i(t) \quad (7)$$

At every time step the constraints on $p^i(t)$ are enforced thereby ensuring that $\boldsymbol{p}(t)$ is a valid probability distribution at all times.

## 4 Shared Control

The shared control paradigm implemented on our robot is a blending-based approach in which the final control command issued to the robot is a linear composition of the human control command and an autonomous robot policy. The autonomous control policy generates control command $\boldsymbol{u}_r \leftarrow f_r(\boldsymbol{x})$ where $f_r(\cdot) \in \mathcal{F}_r$, and $\mathcal{F}_r$ is the set of all control behaviors corresponding to different tasks. This set could be derived using a variety of techniques such as *Learning from Demonstrations* (**??**), motion planners (**??**) or navigation functions (**??**). Specifically, let $\boldsymbol{u}_{r,g}$ be the autonomy command associated with goal $g$. The final control command $\boldsymbol{u}$ issued to the robot then is given by

$$\boldsymbol{u} = \alpha \cdot \boldsymbol{u}_{r,g^*} + (1-\alpha) \cdot \boldsymbol{u}_h$$

where $g^*$ is the most confident goal. The blending factor $\alpha$ is a piecewise linear function of the probability $p(g^*)$ associated with $g^*$ and is given by

$$\alpha = \begin{cases} 0 & p(g^*) \leq \rho_1 \\ \frac{\rho_3}{\rho_2 - \rho_1} \cdot p(g^*) \quad \text{if} & \rho_1 < p(g^*) \leq \rho_2 \\ \rho_3 & p(g^*) > \rho_2 \end{cases}$$

with $\rho_i \in [0,1]\ \forall\ i \in [1,2,3]$ and $\rho_2 > \rho_1$. In our implementation, we empirically set $\rho_1 = \frac{1.2}{n_g}, \rho_2 = \frac{1.4}{n_g}$ and $\rho_3 = 0.7$. Each location $\boldsymbol{x} \in \mathbb{R}^3 \times \mathbb{S}^3$ consists of a position component (3 dimensions) and orientation component (represented as a quaternion in 4 dimensions).[4] Each control command $\boldsymbol{u} \in \mathbb{R}^6$ consists of a translational velocity component (3 dimensions) and rotational velocity component expressed as Euler rates (3 dimensions).

The robot control command $\boldsymbol{u}_{r,g}$ is generated using a simple potential field which is defined in all parts of the state space (**?**). Every goal $g$ is associated with a potential field $\gamma_g$ which treats $g$ as an attractor and all the other goals in the scene as repellers. For potential field $\gamma_g$, the attractor velocity is given by

$$\dot{\boldsymbol{x}}_r^{attract} = \boldsymbol{x}_g - \boldsymbol{x}_{\boldsymbol{r}}$$

[4] $\mathbb{S}^3$ is the space of all unit quaternions.

where $\boldsymbol{x}_g$ is the location of goal $g$.[5] The repeller velocity is given by

$$\dot{\boldsymbol{x}}_r^{repel} = \sum_{i \in \mathcal{G} \setminus g} \frac{\boldsymbol{x}_{\boldsymbol{r}} - \boldsymbol{x}_{g^i}}{\mu(\left\|\boldsymbol{x}_{\boldsymbol{r}} - \boldsymbol{x}_{g^i}\right\|^2)}$$

where $\dot{\boldsymbol{x}}_r$ indicates the velocity of the robot in the world frame, $\mu$ controls the magnitude of the repeller velocity and $\|\cdot\|$ is the Euclidean norm. The autonomy command is computed as a summation of these attractor and repeller velocities.

$$\boldsymbol{u}_{r,g} = \dot{\boldsymbol{x}}_r^{attract} + \dot{\boldsymbol{x}}_r^{repel}$$

$\gamma_g$ operates in the full six dimensional Cartesian space, and treats position and orientation as independent potential fields.

## 5 Study Methods

In this section, we describe the study methods used to evaluate the efficacy of the disambiguation system.

### 5.1 Participants

For this study eight subjects were recruited (mean age: $31 \pm 11$, 3 males and 5 females). All participants gave their informed, signed consent to participate in the experiment, which was approved by Northwestern University's Institutional Review Board.

### 5.2 Hardware

The experiments were performed using the MICO 6-DoF robotic arm (Kinova Robotics, Canada), specifically

| Control Mappings | | |
|---|---|---|
| **Mode** | **Head Array** | **Joystick** |
| 1 | $v_x$ | $v_x, v_y$ |
| 2 | $v_y$ | $v_x, v_z$ |
| 3 | $v_z$ | $\omega_z, \omega_y$ |
| 4 | $\omega_z$ | $\omega_x$ |
| 5 | $\omega_y$ | — |
| 6 | $\omega_x$ | — |

**Fig. 5** A 2-axis joystick (left) and switch-based head array (center) and their operational paradigms (right). $v$ and $\omega$ indicate the translational and rotational velocities of the end-effector, respectively.

[5] In position space, the '–' operator computes the difference between the goal position and current robot position in $\mathbb{R}^3$. In orientation space, the '–' operator computes the *quaternion difference* between the goal orientation and the current robot orientation.

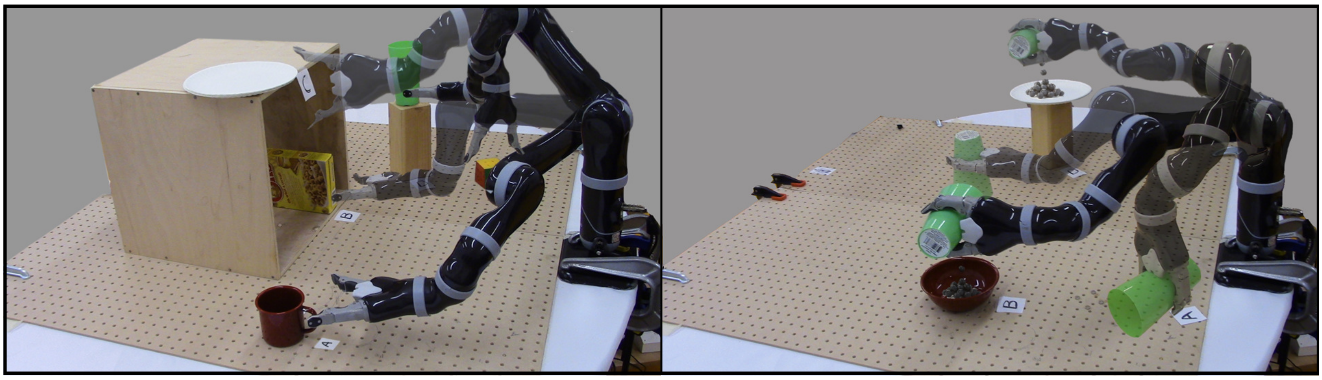

**Fig. 6** Study tasks performed by subjects. *Left:* Single-step reaching task. *Right:* Multi-step Pouring task.

designed for assistive purposes. The software system was implemented using the Robot Operating System (ROS) and data analysis was performed in MATLAB.

The subjects teleoperated the robot using two different control interfaces: a 2-axis joystick and a switch-based head array. The control signals captured from the interfaces were mapped to the Cartesian velocities of the end-effector (Figure 5).

The joystick generated continuous control signals and allowed for control of a maximum of two dimensions at a time. The 6-D control space was partitioned into four control modes that could be accessed using the buttons on the joystick interface. The switch-based head array consisted of three switches embedded in the headrest and generated 1-D discrete signals. The switch at the back of the headrest was used to cycle between the different control modes, and the switches to the left and right controlled the motion of the robot's end effector in the positive and negative directions along the dimension corresponding to the selected control mode. All switches were operated via head movements. An external button was provided to request the mode switch assistance. For both control interfaces the gripper had a dedicated control mode.

#### *5.2.1 Switching Paradigms*

Two kinds of mode switching paradigms were evaluated in the study. Note that the blending assistance was always active for both paradigms. Under the blending paradigm, the amount of assistance was directly proportional to the probability of the most confident goal $g^*$, and thus to the strength of the intent inference. Therefore, if intent inference improved as a result of goal disambiguation, more assistance would be provided by the robot. All trials started in a randomized initial control mode and robot position.

*Manual*: During task execution the user performed all mode switches.

*Disambiguation*: The user additionally could request a disambiguation mode switch at any time during task execution. Upon disambiguation request, the algorithm identified and switched the current control mode to the best disambiguating mode $m^*$. The user was required to request disambiguation at least once during the task execution.

### 5.3 Study Protocol

A within-subjects study was conducted using a fractional factorial design in which the manipulated variables were the tasks, control interfaces and the switching paradigm conditions. Each subject underwent an initial training period that lasted approximately thirty minutes after which the subject performed both tasks using both interfaces under the *Manual* and *Disambiguation* paradigms. The trials were balanced and the control interfaces and the paradigms were randomized and counterbalanced across all subjects to avoid ordering effects. Three trials were collected for the *Manual* paradigm and five trials for the *Disambiguation* paradigm.

**Training:** The training period consisted of three phases and two different task configurations. The subjects used both interfaces to perform the training tasks.

*Phase One*: The subjects were asked to perform a simple reaching motion towards a single goal in the scene. This phase was intended for the subjects to get familiarized with the control interface mappings and teleoperation of the robotic arm.

*Phase Two*: In the second phase of training, the blending-based shared autonomy was introduced. The subjects experienced how the autonomy could provide assistance during the task execution. The subjects were informed that the robot autonomy would be present for all trials of the experiment.

*Phase Three*: For the third phase of the training, multiple objects were introduced in the scene. Subjects were informed that the robot had the capability to pick a

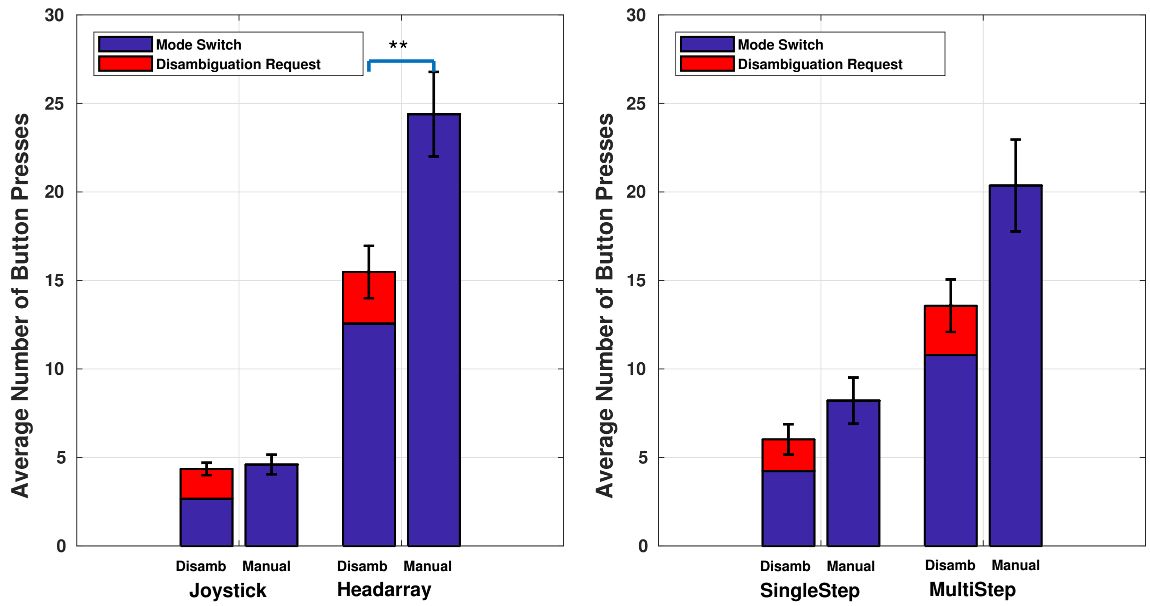


**Fig. 7** Comparison of average number of button presses between *Disambiguation* and *Manual* Paradigms. *Left:* Grouped by control interfaces. *Right:* Grouped by tasks.

control mode that would help it figure out which goal they were going for, and that the subject had the option to request the robot to pick that control mode. Subjects were able to explore this request feature during a reaching task, and were instructed to move as much as s/he can in the control mode chosen by the robot and observe the effects of the mode switch request.

**Testing:** Two different task types were evaluated during testing. Both control interfaces were employed, for all tasks.

*Single-step:* The task is to reach one of five objects on the table, each with a target orientation (Figure 6, Left).

*Multi-step:* Each trial began with a full cup held by the robot gripper. The task required first that the contents of the cup be poured into one of two containers and then that the cup be placed at one of the two specified locations and with a specific orientation (Figure 6, Right).

**Metrics:** The objective metrics used for evaluation include the following. *Number of mode switches* refers to the number of times a user switched between various control modes during task execution. *Number of disambiguation requests* refers to the number of times user pressed the button requesting disambiguating assistance. *Number of button presses* is the sum of *Number of mode switches* and *Number of disambiguation requests*, and is also an indirect measure of user effort. *Onset of robot assistance* refers to the earliest time at which blending assistance to became active. We also characterize the temporal distribution of disambiguation requests.

## 6 Results

Here we report the results of our subject study. Statistical significance was determined by the Wilcoxon Rank-Sum test in where (***) indicates $p < 0.001$, (**) $p < 0.01$ and (*) $p < 0.05$.

### 6.1 Impact of Disambiguation on Task Performance

Figure 7 presents the number of button presses under each mode switching paradigm, for both interfaces and tasks.

A statistically significant decrease in the number of button presses was observed between the *Manual* and *Disambiguation* paradigms when using the headarray (Figure 7, Left). Due to the low-dimensionality of headarray and cyclical nature of mode switching, the number of button presses required for task completion is inherently high. That the disambiguation paradigm was helpful in reducing the number of button presses likely is due to the fact that robot assistance was more effective in the disambiguating control mode and therefore reduced the need for subsequent user-initiated mode switches.

For joystick, statistically significant differences were observed for the number of mode switches between the two paradigms ($p < 0.05$, not pictured in Figure 7). However, the gain due to the reduction of user-initiated mode switches was offset by the button presses that were required for disambiguation requests.

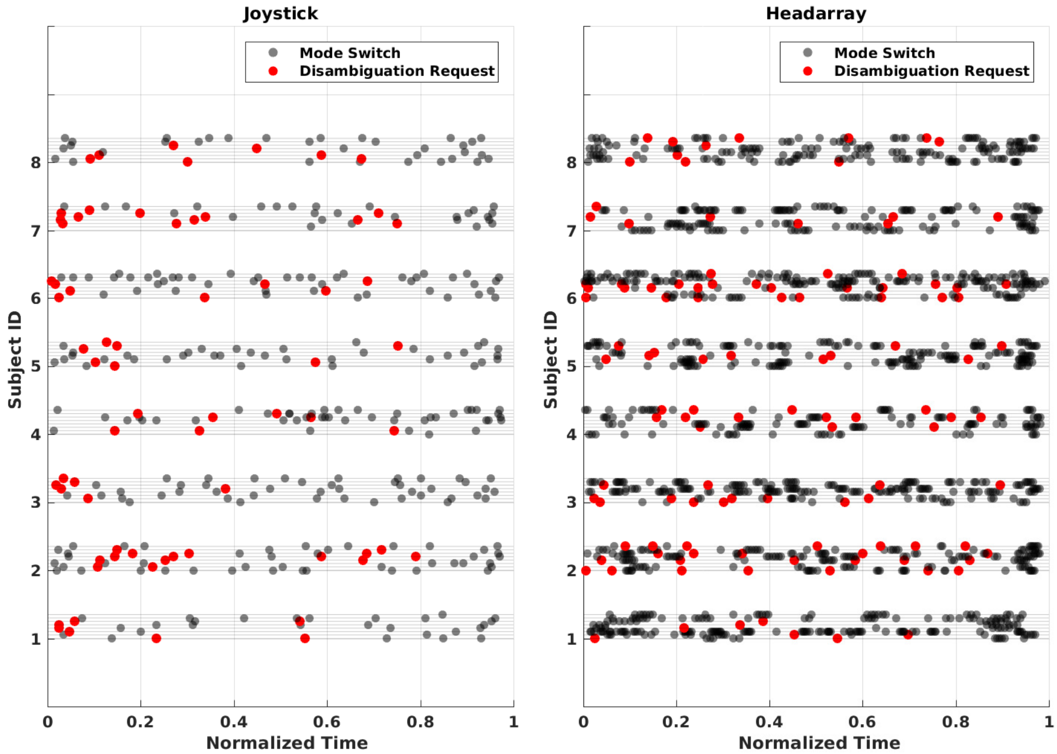


**Fig. 8** Temporal pattern of button presses for each interface and the multi-step task on a trial-by-trial basis for all subjects. Eight trials per subject per interface/task combination.

**Table 1** Characterization of the temporal distribution of disambiguation requests. The values in the table denote the deviation of the temporal distribution from a uniform distribution. Smaller values mean more concentrated and earlier disambiguation requests.

| | Single Step | Multi Step |
|---|---|---|
| Joystick | 0.63 | 0.57 |
| Headarray | 0.35 | 0.22 |

A general trend (although not statistically significant) of a decrease in the number of button presses was also observed for the more complex multi-step task (Figure 7, Right).

These results suggest that disambiguation has a greater effect as the control interface becomes more limited and the task becomes more complex.

## 6.2 Temporal Distribution of Disambiguation Requests

We also observed similar trends in the *temporal distribution* of disambiguation requests, where a higher number of disambiguation requests correlates with the more limited interface and complex task. The temporal distribution of disambiguation requests refers to *when* the subject requested assistance during the course of a trial. We use a measure of *skewness* to characterize how much the temporal distribution deviates from a uniform distribution.[6] Larger positive values of skewness correlate with more concentrated and earlier disambiguation requests. Table 1 reports the skewness of the temporal distribution of disambiguation requests for different interface-task combinations.

Figure 8 shows the temporal pattern of disambiguation requests and mode switches for the multi-step task on a trial-by-trial basis for all subjects. From the figure it is clear that the frequency and density of button presses (disambiguation requests plus mode switches) are much higher for the more limited control interface. The subjects also demonstrated a diverse range of disambiguation request behavior, for example in regards to both when during the execution requests were made and with what frequency (e.g. Subject 8 versus Subject 2, Joystick). The variation between subjects is likely due to different factors such as the user's comfort in

[6] A uniform temporal distribution corresponds to a trial in which the disambiguation requests are uniformly spread out during the course of task execution. The skewness of a uniform distribution is 0.

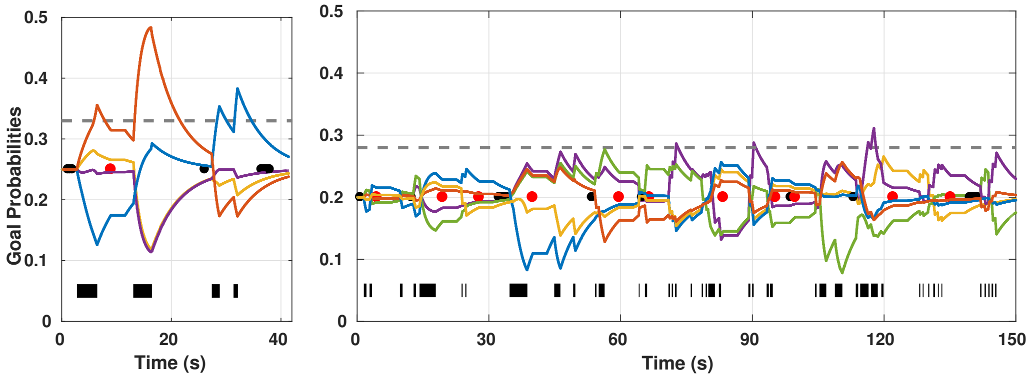


**Fig. 9** Time evolution of goal probabilities. *Top:* Multi-step task. *Bottom:* Single-step task. The gray horizontal line above denotes the minimum threshold for robot assistance. The black bars at the bottom denote non-zero human control commands. The red and black dots indicate button presses that are disambiguation requests and mode switches respectively.

operating the robot and understanding the ability of the disambiguating mode to recruit more assistance from the autonomy.

### 6.3 Onset of Robot Assistance

Our motivating intuition for developing the disambiguation system was that disambiguation should allow the autonomy to step in earlier during the course of task execution. However, our results did not reveal any differences between the two switching paradigms across tasks or across interfaces (Figure 10).

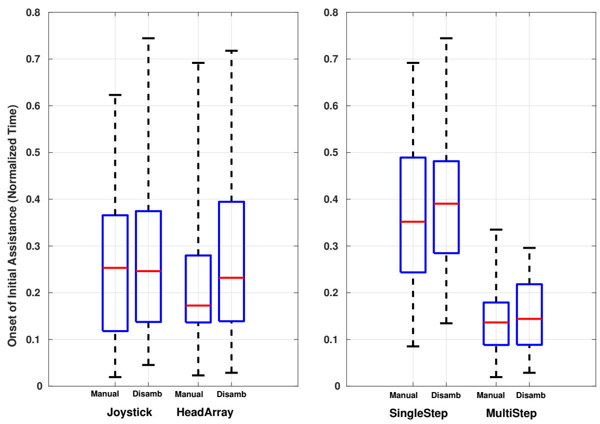


**Fig. 10** Onset of robot assistance normalized with respect to task completion time. *Left:* Across interfaces. *Right:* Across tasks.

One likely contributing factor is that disambiguation frequently was impaired by subjects surprisingly choosing not to operate in the disambiguation mode after making a disambiguation request. The disambiguation power of a control mode depends entirely on the observation of motion within that mode, and so in such scenarios disambiguation is effectively rendered inert.

This phenomenon is illustrated in Figure 9. When control commands are issued, they are indicated by the black bars on the bottom of the plots. Disambiguation requests are shown as red dots, and mode switches as black dots. In the right plot, we see multiple instances of disambiguation requests being followed by no control commands or only very brief control commands. Another interesting behavior is shown in the left plot, where operation in the disambiguating mode very quickly elevates one goal probability above the threshold for providing autonomy assistance, and after the assistance kicks in the subject elects to stop issuing control commands.

## 7 Discussion

In a *help me, help you* type of human robot system, task execution becomes seamless and more efficient when there is a sound mutual understanding of how the other party operates. One surprising observation from our subject study was how often participants submitted a disambiguation request, and then chose not to operate in the selected mode—effectively not helping the robot help them. Of course one possible explanation is that subjects found the disambiguation mode to be in conflict with how they wanted to operate the robot. However another plausible explanation is simply a lack of understanding of how they might help the robot help them. In order to provide *intent-expressive* control commands to the robot, very likely knowledge of the assistance mechanism is critical.

Therefore, the need for extensive and thorough training becomes apparent. The training can be made more effective in a few different ways. First, online feedback of the robot's intent prediction at all times during training can likely help the subject gain a better understanding of the relationship between the characteristics of their control actions (sparsity, aggressiveness, persistence) and the robot's assistive behavior. Second, the subjects could be explicitly informed of the task relevant features (directedness, proximity *et cetera*) that the robot relies on for determining the amount of assistance. Knowledge of these features might motivate the users to leverage the disambiguating mode.

The inherent time delays associated with the computation of the disambiguating mode (approximately 2-2.5s) might have been a discouraging factor and a cause for user frustration. The algorithm could be used to pre-compute a large set of most informative modes for different parts of the workspace, for different goal configurations and for different priors ahead of time, which then might be used a lookup table during task execution. Furthermore, metamodeling techniques and machine learning tools can be used to learn generalizable models that will be effective in previously unseen goal configurations.

In the present system, there is task effort associated with requesting assistance which can discourage the users from utilizing assistance. Automated mode switching schemes can possibly eliminate the need for button presses for disambiguation requests. We also identify an opportunity to have adaptive assistance paradigms that explicitly take into account the characteristics of the user's control behavior. Some users are timid in their operation of the robot whereas some others are more aggressive and confident. Some are more comfortable operating the robot manually and do not seek assistance, whereas some others rely on assistance more frequently. Individual user characteristics could be extracted from training and be used for tuning the parameters of the intent inference engine and the shared control system to maximize robustness and efficacy of the assistive system. This would also likely improve user satisfaction and result in higher user acceptance.

## 8 Conclusion

In this paper, we have presented an algorithm for *intent disambiguation assistance* with a shared-control robotic arm using the notion of *inverse legibility.* The goal of our algorithm is to elicit more *intent-expressive* control commands from the user by placing control in those control modes that *maximally disambiguate* between the various goals in the scene. As a secondary contribution, we also present a novel intent inference mechanism inspired by *dynamic field theory* that works in conjunction with the disambiguation system. Pilot user study was conducted with eight subjects to evaluate the efficacy of the disambiguation system. Our results indicate a decrease in task effort in terms of the number of button presses when disambiguation system employed. In our future work, as informed by our pilot study, we plan to extend the framework into an automated mode switch assistance system. A more extensive user study with motor-impaired subjects will also be conducted in the future to further evaluate the utility of the disambiguation system and explore the disambiguation request patterns of users.

**Biography – Deepak E. Gopinath**

Deepak E. Gopinath is a third-year doctoral student in the Mechanical Engineering Department at Northwestern University and works with Dr. Brenna Argall in the Assistive and Rehabilitation Robotics Laboratory at the Shirley Ryan AbilityLab in Chicago. He completed his B.Tech in Engineering Physics from IIT Bombay in 2007, after which he moved to Boston, USA to pursue a Professional Diploma in Music at Berklee College of Music majoring in Composition and Jazz Performance. Prior to coming to Northwestern, he completed an M.S in Music Technology at Georgia Tech under Dr. Gil Weinberg where he worked in field of Robotic Musicianship. His current research interests are in developing mathematical formalisms for shared-control architectures for assistive robotic manipulators and information theoretic approaches to characterize human-robot interaction.

Bio-Dr. Brenna Argall

Brenna Argall is an Associate Professor of Electrical Engineering & Computer Science, Mechanical Engineering and Physical Medicine & Rehabilitation at Northwestern University. She is also a Faculty Research Scientist at the Shirley Ryan AbilityLab, the premier rehabilitation hosptial in the United States. She is the director of the **a**ssistive & **r**ehabilitation robotics **lab**oratory (**ar**gal**lab**). Her research lies in the intersection of robotics, machine learning and human rehabilitation. The mission of the **ar**gal**lab** is to advance human ability by leveraging autonomy. Dr. Argall is a 2016 recipient of the NSF CAREER award, and her Ph.D. in Robotics (2009) was received from the Robotics Institute at Carnegie Mellon University, as well as her M.S in Robotics (2006) and B.S in Mathematics (2002). Prior to joining Northwestern, she was a postdoctoral fellow (2009-2011) at the École polytechnique fédérale de Lausanne (EPFL), and prior to graduate school she held a Computational Biology position at the National Institutes of Health (NIH).

 

BA - Photo Click here to download Author Photographs BA.png