

PHYSICS-AWARE SOFT-SENSORS FOR EMBEDDED DIGITAL TWINS

Fabio Marcuzzi¹

¹Dipartimento di Matematica “Tullio Levi Civita”,
Università di Padova
marcuzzi@math.unipd.it

iNEST DT course, October 14th 2024



UNIVERSITÀ
DEGLI STUDI
DI PADOVA



DIPARTIMENTO
di MATEMATICA

the innovation will play forever.

SMNUMERICA



Physics-aware soft-sensors

iNEST

Fabio
Marcuzzi

A critical example:

ARIANE 5 First Flight Failure - Flight 501, June 4, 1996, European Space Agency, Rocket Launch.mp4



Here applied mathematics matters:

ARIANE 5

Flight 501 Failure

Report by the Inquiry Board

The Chairman of the Board :

Prof. J. L. LIONS

FOREWORD

On 4 June 1996, the maiden flight of the Ariane 5 launcher ended in a failure. Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 m, the launcher veered off its flight path, broke up and exploded. Engineers from the Ariane 5 project teams of CNES and Industry immediately started to investigate the failure. Over the following days, the Director General of ESA and the Chairman of CNES set up an independent Inquiry Board and nominated the following members :

- Prof. Jacques-Louis Lions (Chairman) Académie des Sciences (France)
- Dr. Lennart L. beck (Vice-Chairman) Swedish Space Corporation (Sweden)
- Mr. Jean-Luc Fauquemergue De l'égalation Générale pour l'Armement (France)
- Mr. Gilles Kahn Institut National de Recherche en Informatique et en Automatique (INRIA), (France)
- Prof. Dr. Ing. Wolfgang Kubbus Technical University of Darmstadt (Germany)
- Dr. Ing. Stephan Lauerbach Daimler-Benz Aerospace (Germany)

there are **restricted data, real-time requirements and technology constraints!**

A few years later, in 2003, Grieves coins the term "Digital Twin" and in 2010 NASA adopts the digital twin philosophy:



Digital Twin: Manufacturing Excellence through Virtual Factory Replication

This paper introduces the concept of a "Digital Twin" as a virtual representation of what has been produced. Compare a Digital Twin to its engineering design to better understand what was produced versus what was designed, tightening the loop between design and execution.

1.4. Top Technical Challenges

Technical Challenges in Priority Order	
1.	Advanced Mission Systems (TABS 4.5); Adaptive Systems
2.	Integrated System Lifecycle Simulation (TABS 3.2); Full Mission Simulation
3.	Simulation-Based Systems Engineering (TABS 3.3); NASA Digital Twin
4.	Software Modeling (TABS 2.1); Formal analysis and traceability of requirements and designs
5.	Integrated Hardware and Software Modeling (TABS 2.2); Advanced Integrated Model & V&V
6.	Modeling (TABS 2.4); Cross-scale and Inter-regional coupling
7.	Flight Computing (TABS 1.1); System Software for Multi-Core Computing
8.	Integrated Hardware and Software Modeling (TABS 2.2); Complexity Analysis Tools
9.	Flight and Ground Computing (TABS 1.1 and 1.2); Eliminate the Multi-core "Programmability Gap"
10.	Software Modeling (TABS 2.1); Software Verification Algorithms



Embedded Digital Twins¹, that is the virtual representations of physical systems that run in embedded systems are deployed on the edge within the embedded software stack to realize e.g. virtual sensors to enrich available information about physical variables and parameters that cannot be provided by direct physical measurements. These lacking measurements are estimated, at least roughly, by an algorithm that processes the available data, usually called a **soft-sensor**.

Let us call a **Physics-aware soft-sensor** the numerical algorithm that performs an indirect measurement by exploiting a **physico-mathematical model** plus a possible **data-driven extension**, used within an **estimation algorithm**.

The complexity of physics-aware soft-sensors depends on:

- ◊ model complexity,
- ◊ interactions with the environment,
- ◊ centrality of measured variables in the virtual measurement process.

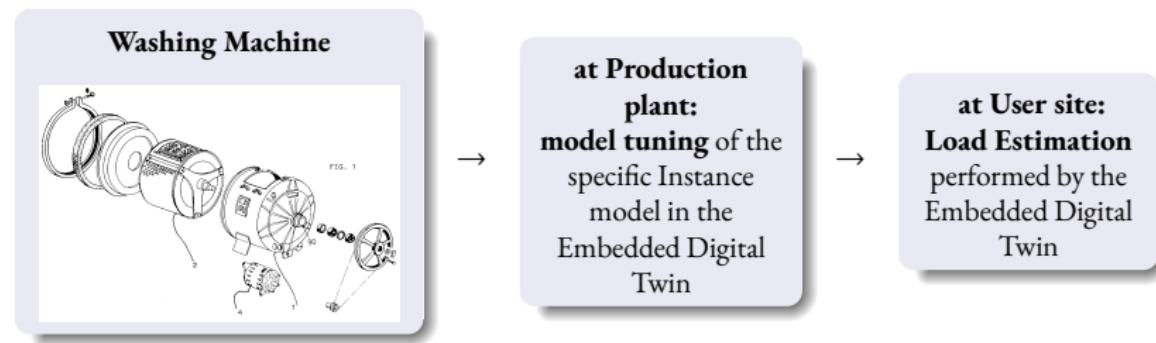
¹Herman Van der Auweraer and Dirk Hartmann. The executable digital twin: merging the digital and the physics worlds, Proceedings ISMA2022, International Conference on Noise and Vibration Engineering, Leuven (B), Sept. 12-14, 2022

Physics-aware soft-sensor I: load estimation inside a washing machine through a spinning process

iNEST

Fabio
Marcuzzi

In this example¹, the microprocessor that controls the machine also run a digital twin of the machine mechanical operation, to estimate the weight of the load inserted by the user. The embedded digital twin of each item has been tuned at the end of the production-line (60 pcs/hour), with a least-squares estimation of physical parameters.



- ◊ no model reduction (physics-based lumped model);
- ◊ online least-squares estimation;
- ◊ 8-bit microcontroller, tenth seconds for computing.

¹European Patent 95112646.5-2314 "Improvement in a washing machine with automatic determination of the weight of the washload": <https://patents.google.com/patent/EP0704568A1/es>

Physics-aware soft-sensor I: load estimation inside a washing machine through a spinning process

iNEST

Fabio
Marcuzzi

European Patent 95112646.5-2314 "Improvement in a washing machine with automatic determination of the weight of the washload" (<https://patents.google.com/patent/EP0704568A1/es>):

(19)		Europäisches Patentamt European Patent Office Office européen des brevets	 (11) EP 0 704 568 A1
EUROPEAN PATENT APPLICATION			
(43) Date of publication:	(51) Int. Cl. ⁶ : D06F 39/00		
03.04.1996 Bulletin 1996/14			
(21) Application number: 95112646.5			
(22) Date of filing: 11.08.1995			
(84) Designated Contracting States: DE ES FR GB IT	(72) Inventors: <ul style="list-style-type: none">• Marcuzzi, Fabio I-33170 Pordenone (IT)• Trangoni, Mario I-33170 Pordenone (IT)• Drius, Francesco I-33052 Cervignano, Udine (IT)		
(30) Priority: 28.09.1994 IT PN940058			
(71) Applicant: ELECTROLUX ZANUSSI ELETTRODOMESTICI S.p.A. I-33170 Pordenone (IT)			

⇒ The textitembedded digital twin can be **patented**, iff there is an innovative physical procedure involved.

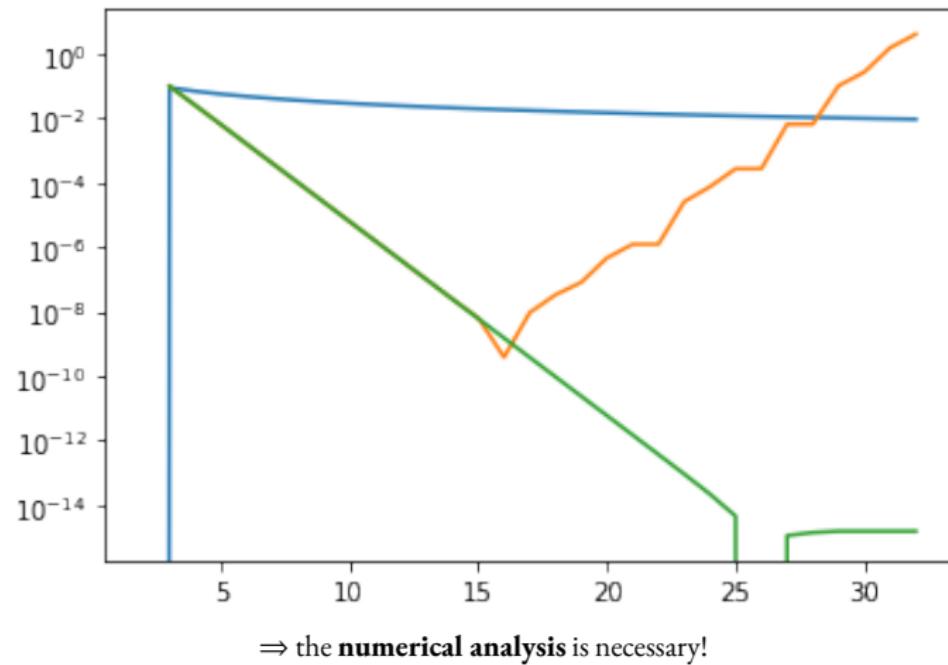
Physics-aware soft-sensor I: load estimation inside a washing machine through a spinning process

iNEST

Fabio

Marcuzzi

An example of numerical instability:

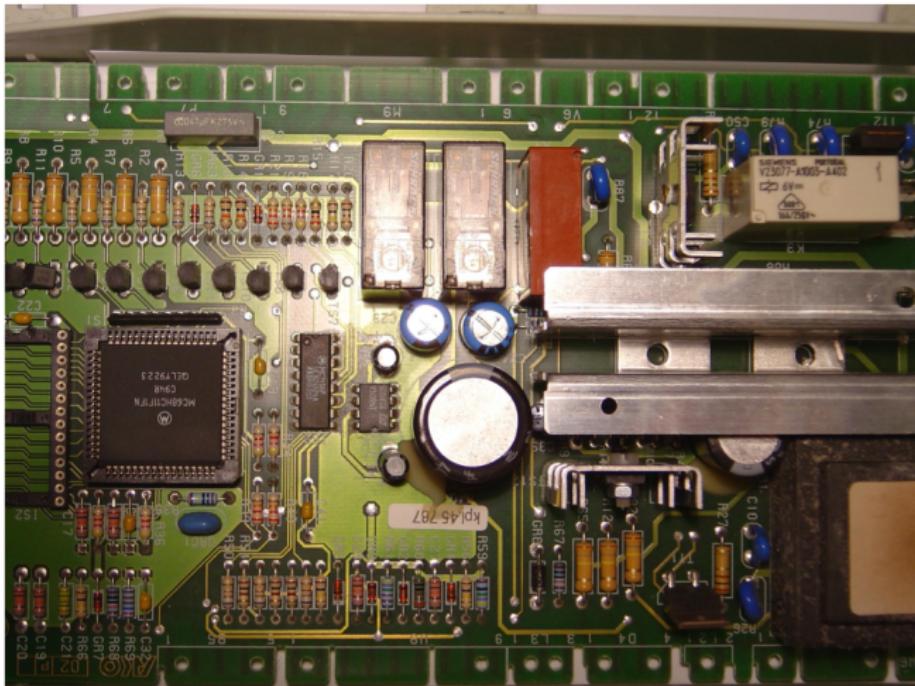


Physics-aware soft-sensor I: load estimation inside a washing machine through a spinning process

iNEST

Fabio
Marcuzzi

In an *embedded digital twin* the numerical analysis is not sufficient, **the algorithm can suffer from the global behaviour of the system**. Examples: a **cold welding**, an NTC with different thermal inertia, etc.

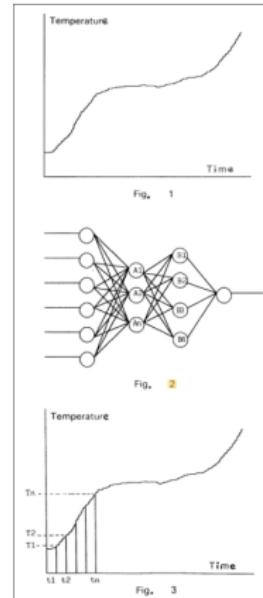


Physics-aware soft-sensor II: drying-time estimation from temperature measurements

iNEST

Fabio

Marcuzzi



Here an important process variable (air humidity) could not be measured, by the machine embedded controller. Therefore, the physics-based model could not be used and the data-driven model (a "**physical-features-informed neural network**") tried to understand the physics from the available data and predict the drying time:

⇒ **prediction over Data Driven model Discovery.**

Modeling characteristics of utmost importance:

poor interpretability of the model,

- ◊ no generalizability,
- ◊ trustworthiness hard to reach (implicit nonlinear surface response),
- ◊ high computational efficiency (12-bit arithmetic were sufficient),
- ◊ no self-adaptation,
- ◊ 8-bit microcontroller, almost **one minute to forward run the neural network.**

¹European Patent Application EP95114450A "Improvement in the arrangement used in a clothes drying apparatus to determine the drying time": <https://patents.google.com/patent/EP0707107A1/pt>

Physics-aware soft-sensor II: drying-time estimation from temperature measurements

European Patent Application EP95114450A "Improvement in the arrangement used in a clothes drying apparatus to determine the drying time": <https://patents.google.com/patent/EP0707107A1>

(19)		Europäisches Patentamt European Patent Office Office européen des brevets	 (11) EP 0 707 107 A1
(12) EUROPEAN PATENT APPLICATION			
(43) Date of publication:			(51) Int. Cl. ⁶ : D06F 58/28
17.04.1996 Bulletin 1996/16			
(21) Application number: 95114450.0			
(22) Date of filing: 14.09.1995			
<hr/> <p>(84) Designated Contracting States: DE ES FR GB IT</p> <p>(30) Priority: 11.10.1994 IT PN940061</p> <p>(71) Applicant: ELECTROLUX ZANUSSI ELETTRODOMESTICI S.p.A. I-33170 Pordenone (IT)</p> <p>(72) Inventors: • Haberl, Ingo I-33170 Pordenone (IT)</p>		<ul style="list-style-type: none">• Marcuzzi, Fabio I-33170 Pordenone (IT)• Drius, Francesco I-33052 Cervignano, Udine (IT) <p>(74) Representative: Giugni, Valter et al PROPRIA Protezione Proprietà Industriale S.r.l. Via Mazzini 13 I-33170 Pordenone (IT)</p>	

⇒ Patent NOT accepted: the experimental settings was been already patented.

Scientific machine learning:

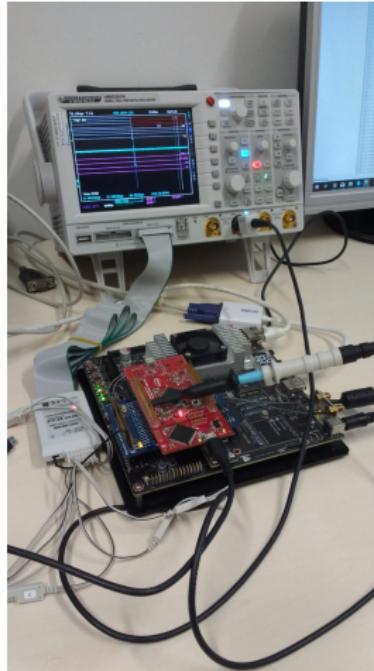
- 1990-2000: Lagaris, Likas and Papageorgiou used neural networks to solve partial differential equations;
- 2015: J.S. Hesthaven, G. Rozza, B. Stamm, "Certified Reduced Basis Methods for Parametrized Partial Differential Equations";
- 2019: Raissi, Perdikaris and Karniadakis, "PINNs (Physics-Informed Neural Networks)";
- 2021: S. Wang, H. Wang and P. Perdikaris, "Learning the solution operator of parametric partial differential equations with physics-informed DeepONets";
- 2023: R. Geelen, S. Wright, K. Willcox, "Operator Inference for non-intrusive model reduction with quadratic manifolds";
- 2023: F. Regazzoni, S. Pagani, M. Salvador, L. Dede', A. Quarteroni, "Latent Dynamics Networks: learning the intrinsic dynamics of spatio-temporal processes"
- 2023: I. Gonnella, M. Hess, G. Stabile, G. Rozza, "A two-stage deep learning architecture for model reduction of parametric time-dependent problems";
- 2024: D. Patel, D. Ray, M. R. Abdelmalik, T. J. Hughes, A. A. Oberai, "Variationally Mimetic Operator Networks"

⇒ Let us see how this scenario is evolving in **embedded real-time computing (edge computing)**.

Embedded computing issues for physics-aware soft-sensors

iNEST

Fabio
Marcuzzi



- ◊ **Numerical accuracy:** from 8-bit microcontrollers, to 32-bit ARM cores (single precision HW f.p.)
- ◊ **Computing power:** clocks of embedded systems is much lower than PCs.
- ◊ limited **software libraries for computing**, e.g. numerical linear algebra.
- ◊ Modelling for physics-aware soft-sensors with differential equations or sophisticated linear algebra implies to **debug** nontrivial numerical models on a real-time system¹.

→ for this reason we developed a tool for **automatic translation** from LaTeX to Python and C, to debug the same code on a PC emulation.

⇒ Moreover, this calls for the implementation of algorithms with a **broad applicability**, like the PAD-NMF is.

¹Dessole M., Marcuzzi F., Fully iterative ILU preconditioning of the unsteady Navier–Stokes equations for GPGPU (2019) Computers and Mathematics with Applications, 77 (4), pp. 907 - 927



iNEST

Fabio
Marcuzzi

For mechanical/civil structures the digital twin model is a Finite Element model:



... and for embedded digital twins ?!?

```

1# SimNumerica uLab © 2010
File Machine Debug procedure Test sequences Simulation Network Help MPLAB-X
[plant]
Details | Parameters and Variables | Pins and Pads | Numerical m...
Python model code:
101 ma_h_hist1 = []
102 ma_h_hist2 = []
103 ma_h_hist3 = []
104
105 x = [float(i) for i in Odysseus.current]
106 a_ext_mean = np.array(x)
107
108 try:
109     t_0
110 except NameError:
111     t_0 = 0
112
113 dtmin = .05
114 dtmax = time_size()
115 for i in range(1, n, T_dtmax):
116     bracket_70 = Function("x[0]*k[11]*")
117     bracket_10 = Function("x[1]*k[0]*")
118     bracket_11 = Function("v[0]*k[11]*")
119     bracket_20 = Function("v[0]*w[0]*k[11]*")
120     bracket_21 = Function("v[1]*w[0]*k[11]*")
121     bracket_22 = Function("v[0]*w[1]*k[11]*")
122     bracket_23 = Function("v[1]*w[1]*k[11]*")
123     bracket_24 = Function("v[0]*k[11]*")
124     bracket_25 = Function("v[1]*k[11]*")
125
126     l1=term(x[4]) == ma_h_hist0_init, va
127     l2=term(x[4]) == ma_h_hist1 + ma_h_hist2, va
128     l3=term(x[4]) == ma_h_hist3, va
129     l4=term(x[4]) == ma_h_hist0, va
130
131     pr = Problem.current[0]
132     pr.method.set_t_max("10sec", "Time")
133     pr.method.used_functions.update(["v[0]", "v[1]", "w[0]", "w[1]"])
134
135     x = pr.method.activate(t)
136
137     t += dt
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
517
518
519
519
520
521
522
523
524
525
526
527
527
528
529
529
530
531
532
533
534
535
536
536
537
538
538
539
539
540
541
542
543
544
545
545
546
547
547
548
548
549
549
550
551
552
553
554
555
555
556
557
557
558
558
559
559
560
561
562
563
563
564
565
565
566
566
567
567
568
568
569
569
570
571
572
573
573
574
575
575
576
576
577
577
578
578
579
579
580
581
582
583
583
584
585
585
586
586
587
587
588
588
589
589
590
591
592
593
593
594
595
595
596
596
597
597
598
598
599
599
600
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1600
1601
1601
1602
1602
1603
1603
1604
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1610
1611
1611
1612
1612
1613
1613
1614
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1620
1621
1621
1622
1622
1623
1623
1624
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1630
1631
1631
1632
1632
163
```

Access Automation systems: Electric Sliding Gate

iNEST

Fabio
Marcuzzi

The first step for a digital twin is the construction of a mathematical model:

Mathematical Equations (no FOM available):

a DC motor with its drive, reduction gear,
pinionrack system, gate:

$$J_{eq} \frac{d\omega_m(t)}{dt} + B_{eq}\omega_m(t) = \tau_m(t) - F(t) \frac{N_1 N_g}{\eta_1 \eta_g}$$

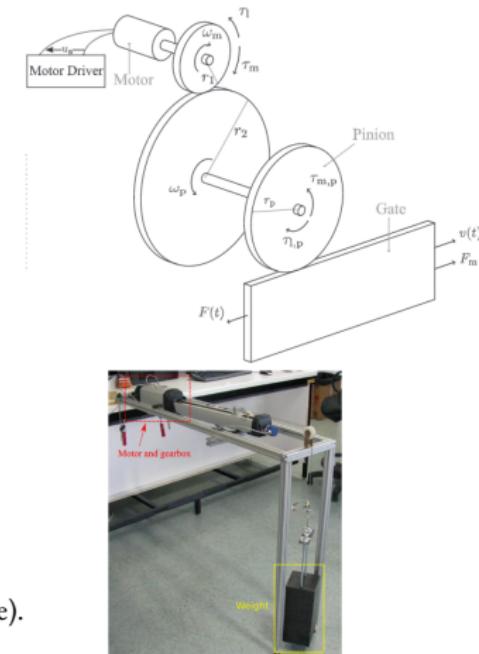
with

$$J_{eq} = J_m + J_p \frac{N_g^2}{\eta_g} + M \frac{N_1^2 N_g^2}{\eta_1 \eta_g}$$

$$B_{eq} = B_m + B_c \frac{N_1^2 N_g^2}{\eta_1 \eta_g}$$

Computational inverse problem¹²:

System Identification and physical parameters estimation (offline).



¹Marcuzzi, Fabio (2018). Linear estimation of physical parameters with subsampled and delayed data. JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS, vol. 331, p. 11-22

²Chiara Faccio, Fabio Marcuzzi (2022). A linear algorithm for the minimal realization problem in physical coordinates with a non-invertible output matrix. LINEAR ALGEBRA AND ITS APPLICATIONS, vol. 644, p. 149-171

A sensor can detect an effect, but a digital twin, thank to the mathematical model of the system, can describe the **dependencies** and connect **causes and consequences** of the observed phenomena.

Access Automation systems: Electric Sliding Gate¹

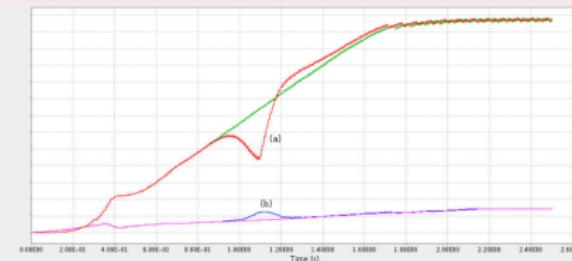
Delayed opening: the opening of the gate took more time than expected, why?

unexpected friction on the transmission

irregular generation of the motor torque

Temporary exogenous friction: the digital twin compares the predicted and the measured behaviours (see Figure from¹) and, using the mathematical model, can estimate the exogenous cause

of the discrepancy.²



green: predicted speed, red: measured speed, magenta: predicted motor current, blue: measured motor current.

¹A. Beghi, F. Marcuzzi, P. Martin, F. Tinazzi and M. Zigliotto, Virtual prototyping of embedded control software in mechatronic systems: A case study. *Mechatronics* (2017) **43**:99–111.



The mathematical model for a digital twin can be used preliminarily to implement and verify the control software in a virtual prototype and then to transfer the control software directly into the physical system:

barriera che si muove con firmware sviluppato con muLab.mp4



and its present version:

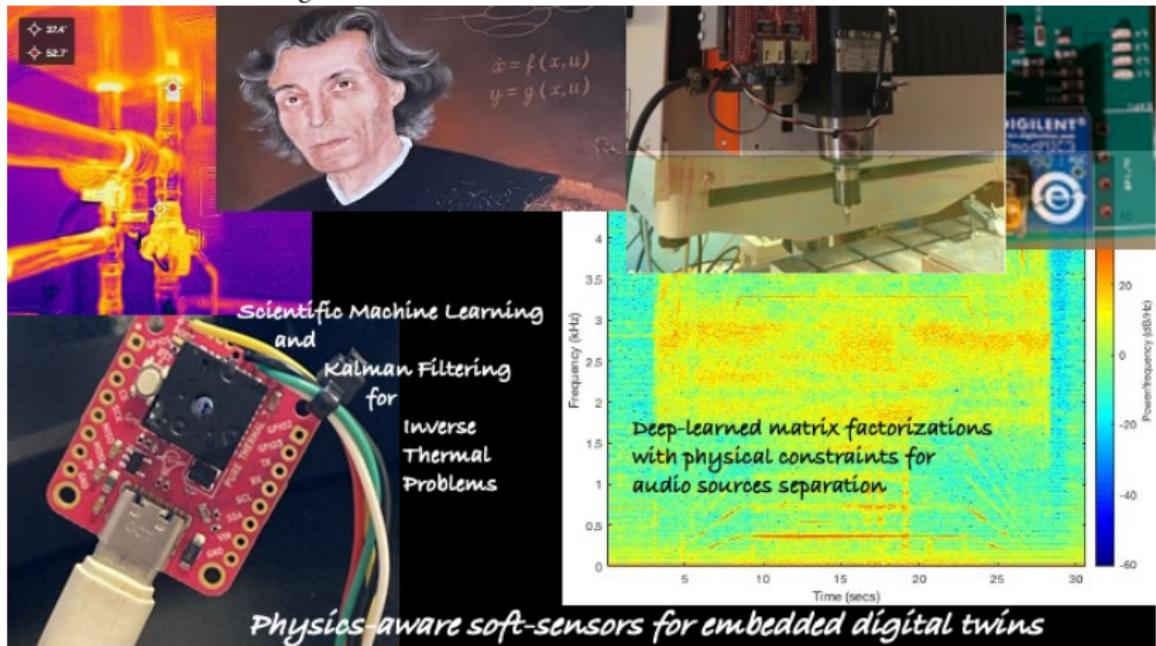
eCfL Digital Twin example.mp4

Recent, multi-disciplinary convergence in embedded computing

iNEST

Fabio
Marcuzzi

New sensors and 32-bit CPUs with HW FP units **empower new algorithms** based on systems theory, scientific machine learning, ...



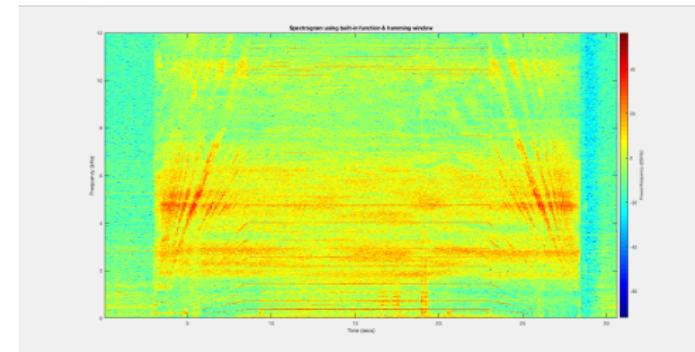
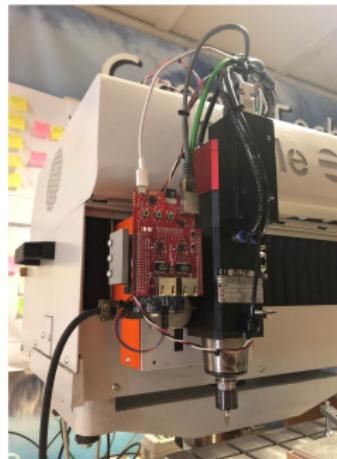
In experimental vibration analysis, using a **microphone** instead of the usual accelerometer, **the goal is to separate the acoustic created by the process to be monitored $\{s_t\}$, from that generated by the environment $\{n_t\}$.**

The quantity we measure is a mixture $\{x_t\}$, given by the sum of the two sources:

$$x_t = s_t + n_t$$

Taking the elementwise squared norm of the STFT matrices for the above signals, yields an additive decomposition into matrix spectrograms:

$$X \approx S + N$$





Deep-NMF¹: Introduction

iNEST

Fabio
Marcuzzi

Let us consider a regularized NMF problem in the form

$$\text{minimize} \quad D_1(X, WH) + \square \|H\|_1$$

$$\text{subject to} \quad W \in \mathcal{M}_{M \times R}(\mathbb{R}^+) \quad , \quad H \in \mathcal{M}_{R \times N}(\mathbb{R}^+)$$

which is tackled by alternate optimization of the factors

$$H^{(k+1)} = f(X; W^{(k)}, H^{(k)}) \quad , \quad W^{(k+1)} = g(X; W^{(k)}, H^{(k+1)})$$

¹Hershey J. R., Le Roux J., Weninger F. - *Deep Unfolding: Model-Based Inspiration of Novel Deep Architectures*, (2014)

Deep-NMF¹: Introduction

Let us consider a regularized NMF problem in the form

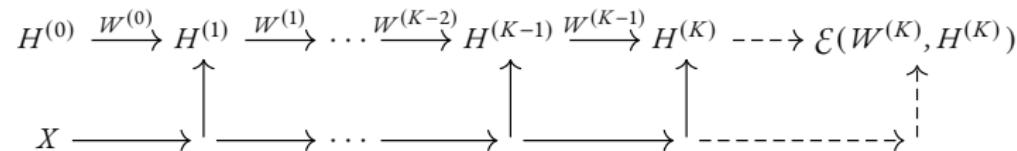
$$\text{minimize} \quad D_1(X, WH) + \square \|H\|_1$$

$$\text{subject to} \quad W \in \mathcal{M}_{M \times R}(\mathbb{R}^+) \quad , \quad H \in \mathcal{M}_{R \times N}(\mathbb{R}^+)$$

which is tackled by alternate optimization of the factors

$$H^{(k+1)} = f(X; W^{(k)}, H^{(k)}) \quad , \quad W^{(k+1)} = g(X; W^{(k)}, H^{(k+1)})$$

By **interpreting the iterative update scheme as a neural network**, where $H^{(k+1)}$ is the output of the k -th layer given the input $H^{(k)}$ and activation function f , Deep-NMF tries to address the convergence issue typical of NMF by *untangling* the bases across layers.



¹Hershey J. R., Le Roux J., Weninger F. - *Deep Unfolding: Model-Based Inspiration of Novel Deep Architectures*, (2014)

Deep-NMF¹: Introduction

Let us consider a regularized NMF problem in the form

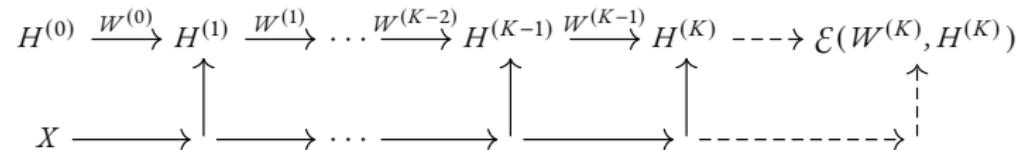
$$\text{minimize} \quad D_1(X, WH) + \square \|H\|_1$$

$$\text{subject to} \quad W \in \mathcal{M}_{M \times R}(\mathbb{R}^+) \quad , \quad H \in \mathcal{M}_{R \times N}(\mathbb{R}^+)$$

which is tackled by alternate optimization of the factors

$$H^{(k+1)} = f(X; W^{(k)}, H^{(k)}) \quad , \quad W^{(k+1)} = g(X; W^{(k)}, H^{(k+1)})$$

By **interpreting the iterative update scheme as a neural network**, where $H^{(k+1)}$ is the output of the k -th layer given the input $H^{(k)}$ and activation function f , Deep-NMF tries to address the convergence issue typical of NMF by *untangling* the bases across layers.



Denoting S the clean source spectrogram we want to separate and reconstruct from the mixture, $W_S^{(K)}$ the subdictionary allocated to describe clean frequencies during the training phase and $H_S^{(K)}$ the corresponding rows in the coefficient's matrix, a possible choice for the loss-function \mathcal{E} is

$$\mathcal{E} = \|S - F \circ X\|_2^2 \quad , \quad F = \frac{W_S^{(K)} H_S^{(K)}}{W^{(K)} H^{(K)}}$$

¹Hershey J. R., Le Roux J., Weninger F. - *Deep Unfolding: Model-Based Inspiration of Novel Deep Architectures*, (2014)



Deep-NMF: A peek under the hood

iNEST

Fabio
Marcuzzi

The backpropagation algorithm is non-conventional, since the weights $W^{(k)}$ **must remain nonnegative...**

Deep-NMF: A peek under the hood

iNEST

Fabio
Marcuzzi

The backpropagation algorithm is non-conventional, since the weights $W^{(k)}$ **must remain nonnegative**...

...so we split the gradients into positive/negative part and recursively compute $[\nabla_{H^{(k)}} \mathcal{E}]_{\pm}$...

$$\begin{aligned} [\nabla_{H^{(k)}} \mathcal{E}]_{\pm} &= \frac{W^{(k)\top} \frac{X}{W^{(k)} H^{(k)}}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\pm} \\ &\quad + W^{(k)\top} \left(\frac{X}{(W^{(k)} H^{(k)})^{\circ 2}} \circ \left(W^{(k)} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\mp} \right) \right) \right) \end{aligned}$$

Deep-NMF: A peek under the hood

iNEST

Fabio
Marcuzzi

The backpropagation algorithm is non-conventional, since the weights $W^{(k)}$ **must remain nonnegative**...

...so we split the gradients into positive/negative part and recursively compute $[\nabla_{H^{(k)}} \mathcal{E}]_{\pm}$...

$$[\nabla_{H^{(k)}} \mathcal{E}]_{\pm} = \frac{W^{(k)\top} \frac{X}{W^{(k)} H^{(k)}}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\pm} \\ + W^{(k)\top} \left(\frac{X}{(W^{(k)} H^{(k)})^{\circ 2}} \circ \left(W^{(k)} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\mp} \right) \right) \right)$$

...in order to obtain $[\nabla_{W^{(k)}} \mathcal{E}]_{\pm}$...

$$[\nabla_{W^{(k)}} \mathcal{E}]_{\pm} = \frac{X}{W^{(k)} H^{(k)}} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\pm} \right)^{\top} \\ + \left(\frac{X}{(W^{(k)} H^{(k)})^{\circ 2}} \circ \left(W^{(k)} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\mp} \right) \right) \right) H^{(k)\top} \\ + \mathbb{1}_{N \times M} \left(\frac{H^{(k)}}{(W^{(k)\top} \mathbb{1}_{N \times M} + \square)^{\circ 2}} \circ \left(W^{(k)\top} \frac{X}{W^{(k)} H^{(k)}} \right) \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\mp} \right)^{\top}$$

Deep-NMF: A peek under the hood

iNEST

Fabio
Marcuzzi

The backpropagation algorithm is non-conventional, since the weights $W^{(k)}$ **must remain nonnegative**...

...so we split the gradients into positive/negative part and recursively compute $[\nabla_{H^{(k)}} \mathcal{E}]_{\pm}$...

$$[\nabla_{H^{(k)}} \mathcal{E}]_{\pm} = \frac{W^{(k)\top} \frac{X}{W^{(k)} H^{(k)}}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\pm} \\ + W^{(k)\top} \left(\frac{X}{(W^{(k)} H^{(k)})^{\circ 2}} \circ \left(W^{(k)} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{+} \right) \right) \right)$$

...in order to obtain $[\nabla_{W^{(k)}} \mathcal{E}]_{\pm}$...

$$[\nabla_{W^{(k)}} \mathcal{E}]_{\pm} = \frac{X}{W^{(k)} H^{(k)}} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{\pm} \right)^{\top} \\ + \left(\frac{X}{(W^{(k)} H^{(k)})^{\circ 2}} \circ \left(W^{(k)} \left(\frac{H^{(k)}}{W^{(k)\top} \mathbb{1}_{N \times M} + \square} \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{+} \right) \right) \right) H^{(k)\top} \\ + \mathbb{1}_{N \times M} \left(\frac{H^{(k)}}{(W^{(k)\top} \mathbb{1}_{N \times M} + \square)^{\circ 2}} \circ \left(W^{(k)\top} \frac{X}{W^{(k)} H^{(k)}} \right) \circ [\nabla_{H^{(k+1)}} \mathcal{E}]_{+} \right)^{\top}$$

...and finally update the weights.

$$W^{(k)} \Leftarrow W^{(k)} \circ \frac{[\nabla_{W^{(k)}} \mathcal{E}]_{-}}{[\nabla_{W^{(k)}} \mathcal{E}]_{+}}$$



Physics-aware Deep-NMF²

iNEST

Fabio
Marcuzzi

The Deep-NMF algorithm and architecture was specifically designed for speech enhancement, which effectively makes it ill-suited to deal with datasets stemming from *physico-mathematical models*, where the clean spectrograms S have strong **intrinsic structure**.

We proposed several **physics-aware enhancements**:

²Erik Chinellato and Fabio Marcuzzi. Hits detection in audio mixtures by means of a physics-aware Deep-NMF algorithm, in review, 2024.

Physics-aware Deep-NMF²

iNEST

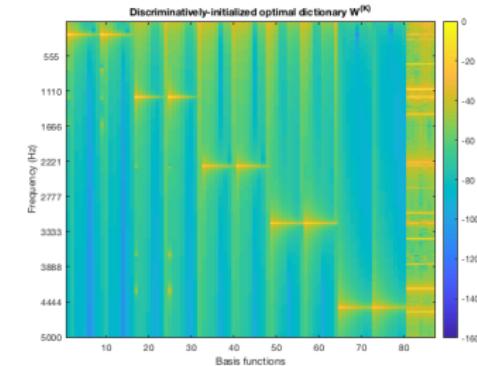
Fabio

Marcuzzi

The Deep-NMF algorithm and architecture was specifically designed for speech enhancement, which effectively makes it ill-suited to deal with datasets stemming from *physico-mathematical models*, where the clean spectrograms S have strong **intrinsic structure**.

We proposed several **physics-aware enhancements**:

- Creation of an **optimal discriminative dictionary** $W^{(K)} = \hat{W}$ (not modified by back-propagation) which enables the recognition of physically-characterized hit responses;



²Erik Chinellato and Fabio Marcuzzi. Hits detection in audio mixtures by means of a physics-aware Deep-NMF algorithm, in review, 2024.

Physics-aware Deep-NMF²

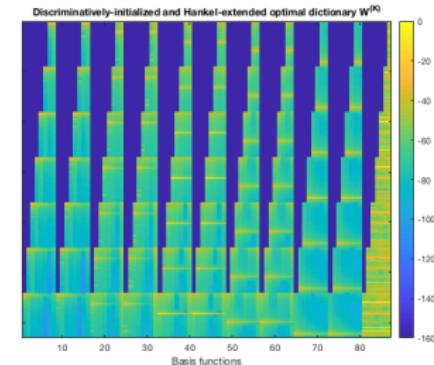
iNEST

Fabio
Marcuzzi

The Deep-NMF algorithm and architecture was specifically designed for speech enhancement, which effectively makes it ill-suited to deal with datasets stemming from *physico-mathematical models*, where the clean spectrograms S have strong **intrinsic structure**.

We proposed several **physics-aware enhancements**:

- Creation of an **optimal discriminative dictionary** $W^{(K)} = \hat{W}$ (not modified by back-propagation) which enables the recognition of physically-characterized hit responses;
- Embedding of the time-correlation between consecutive frames in the spectrograms by employing **block-Hankel** weights $W^{(k)}$;



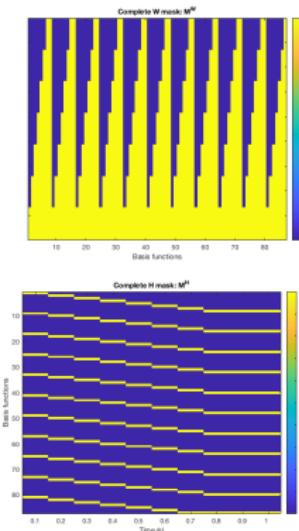
²Erik Chinellato and Fabio Marcuzzi. Hits detection in audio mixtures by means of a physics-aware Deep-NMF algorithm, in review, 2024.

Physics-aware Deep-NMF²

The Deep-NMF algorithm and architecture was specifically designed for speech enhancement, which effectively makes it ill-suited to deal with datasets stemming from *physico-mathematical models*, where the clean spectrograms S have strong **intrinsic structure**.

We proposed several **physics-aware enhancements**:

- Creation of an optimal discriminative dictionary $W^{(K)} = \hat{W}$ (not modified by back-propagation) which enables the recognition of physically-characterized hit responses;
- Embedding of the time-correlation between consecutive frames in the spectrograms by employing **block-Hankel weights** $W^{(k)}$;
- Preservation of the block-Hankel structure by **projection**, thus modifying both the forward- and back-propagation;



²Erik Chinellato and Fabio Marcuzzi. Hits detection in audio mixtures by means of a physics-aware Deep-NMF algorithm, in review, 2024.

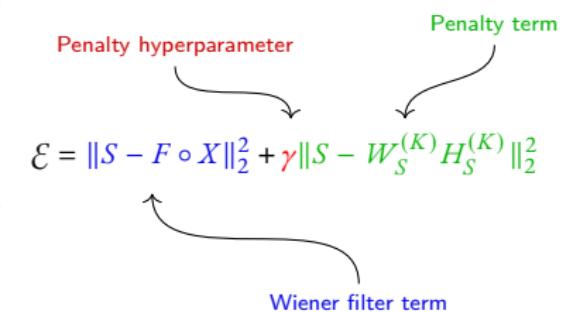
The Deep-NMF algorithm and architecture was specifically designed for speech enhancement, which effectively makes it ill-suited to deal with datasets stemming from *physico-mathematical models*, where the clean spectrograms S have strong **intrinsic structure**.

We proposed several **physics-aware enhancements**:

- Creation of an **optimal discriminative dictionary** $W^{(K)} = \hat{W}$ (not modified by back-propagation) which enables the recognition of physically-characterized hit responses;
- Embedding of the time-correlation between consecutive frames in the spectrograms by employing **block-Hankel weights** $W^{(k)}$;
- Preservation of the block-Hankel structure by **projection**, thus modifying both the forward- and back-propagation;
- Construction of a **better-suited loss function** for the training process, enforcing a more accurate reconstruction of the clean spectrogram.

$$\mathcal{E} = \|S - F \circ X\|_2^2 + \gamma \|S - W_S^{(K)} H_S^{(K)}\|_2^2$$

Penalty hyperparameter Penalty term
Wiener filter term



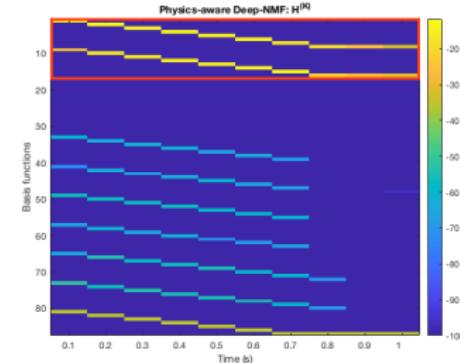
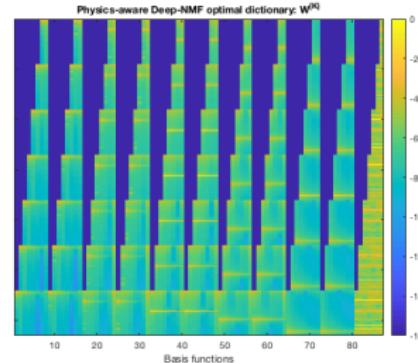
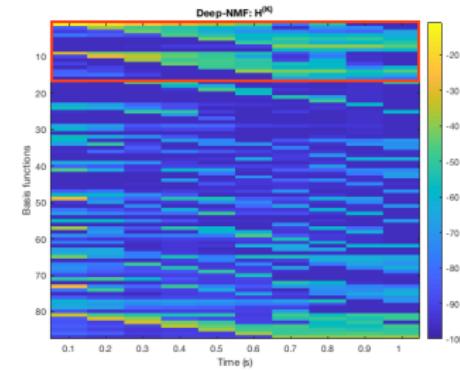
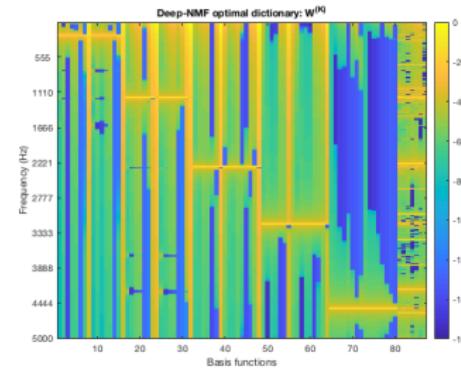
²Erik Chinellato and Fabio Marcuzzi. Hits detection in audio mixtures by means of a physics-aware Deep-NMF algorithm, in review, 2024.

Physics-aware soft-sensor III: hits detection from audio source-separation

iNEST

Fabio

Marcuzzi



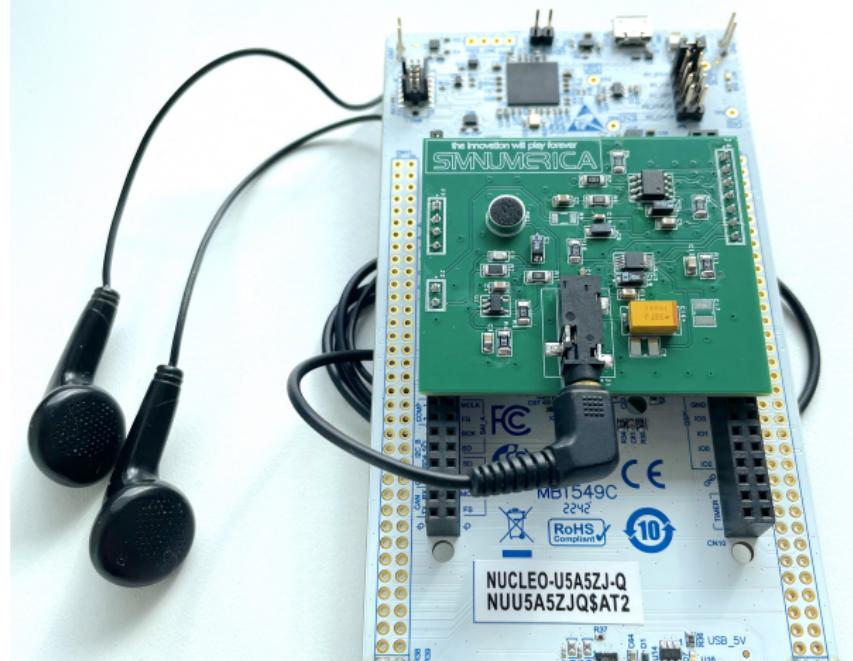
In order to process 1 second of mixture signal and produce the Physics-aware Deep-NMF output matrix $H^{(K)}$, for the above example around 5.5 million floating point operations are needed. As a consequence, an **ARM4 microcontroller** (144 MHz, with a HW single-precision floating-point unit) with its ~ 8 MFLOPS, can **run the complete algorithm** in less than **0.7 seconds**.

The IOT system

iNEST

Fabio

Marcuzzi



Computing resources vs computational cost

iNEST

Fabio
Marcuzzi

Denoting:

M : total number of frequency buckets

N : time instants resulting from applying the STFT to 1 sec long signal mixtures,

K : the number of layers in the network,

R : the number of columns in each PAD-NMF dictionary $W^{(k)}$,

T : the number of Hankel concatenations,

the computational cost of a forward run of the PAD-NMF algorithm is:

$$C_{\text{PAD-NMF}} = O(MTR + (K - 1)(MTRN + MTN + 2RN + 1))$$

Note that $(M, N) = \text{size}(X)$.

Computing resources vs computational cost

Furthermore, letting N_S be the number of sources we are trying to detect and reconstruct, $R_n \forall n = 1, \dots, N_S$ the number of columns allocated within the dictionaries to describe the corresponding source, and M_{IT} the number of optimization iterations employed to solve problem (??), the computational cost of the hit detection procedure is:

$$C_{\text{HitDet}} = O\left(N_S(6N_S + 7)M_{\text{IT}} + N_S + \sum_{n=1}^{N_S} (R_n + N - 1)(R_n + 1)\right)$$

Lastly, denoting n_X the number of samples of the discrete-time mixture signals (for 1s mixtures $n_X = f_s$, where f_s is the sampling frequency) and W the window length used in the STFT expressed in number of samples, the clean audio signal generation has a computational cost given by:

$$C_{\text{SigGen}} = O\left(n_X + 2N(W + 1) \sum_{n=1}^{N_S} R_n\right)$$

To be run online, execution time has to remain below 1s. Assuming $N_S = 3$ and $n_X = 44100$:

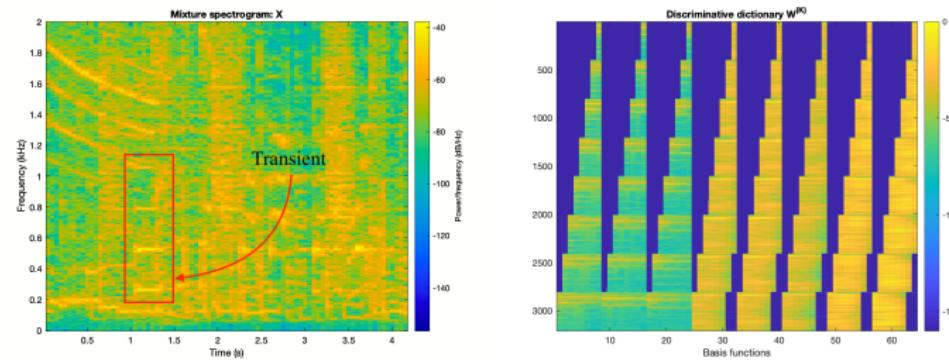
An ARM4 micro controller, given its ~ 8 MFLOPS, allows us to let $M = 143$, $N = T = 5$, $K = 4$, $R = 55$, $R_n \equiv 5$, $M_{\text{IT}} = 5000$ and $W = 22050$ for a total execution time of 0.8641s.

An ARM7 micro controller, given its ~ 20 MFLOPS, allows us to let $M = 143$, $N = T = 8$, $K = 4$, $R = 88$, $R_n \equiv 8$, $M_{\text{IT}} = 5000$ and $W = 17640$ for a total execution time of 0.8495s.

Experiment: piano notes and a loud music

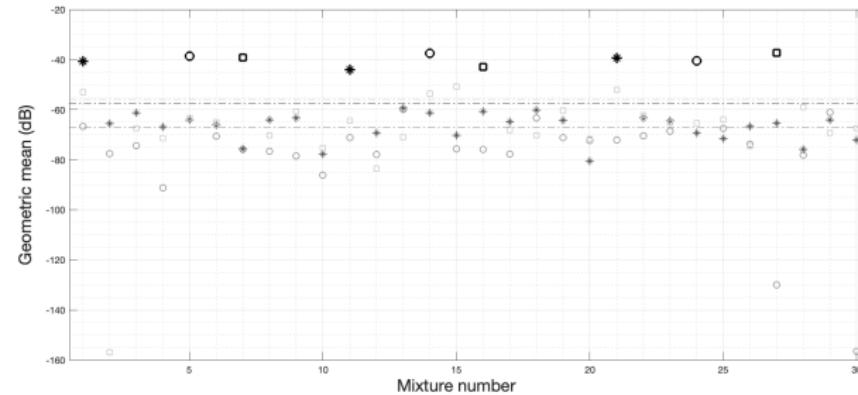
iNEST

Fabio
Marcuzzi

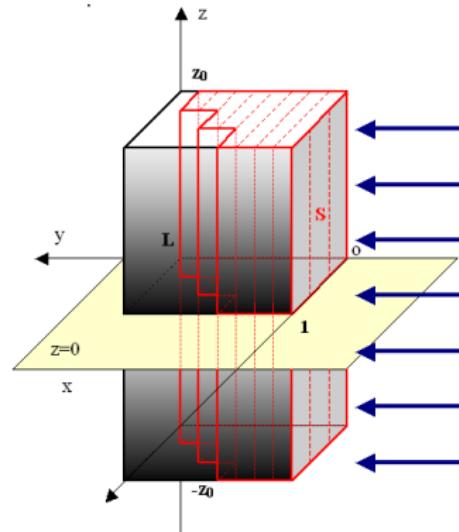


Experiment: piano notes and a loud music

iNEST

Fabio
Marcuzzi

² E. Chinellato, F. Marcuzzi, P. Martin. Real-Time Generation of a Targeted Clean Audio Sequence from Source Separation of Noisy Environmental Mixtures Using a Deep Nonnegative Matrix Factorization on IOT Devices. ICT for Intelligent Systems. ICTIS 2024. Smart Innovation, Systems and Technologies, vol 404. Springer, Singapore. https://doi.org/10.1007/978-981-975810-4_23.



Aim: to estimate the inner material change without solving a nonlinear shape estimation problem.

We demonstrated¹ that there exists a fictitious heat source $f_{\mathcal{S}}(t, x, y)$

such that for the solution $T_{(f_{\mathcal{S}})}$ of

$$\begin{cases} \rho C_p T_{(f_{\mathcal{S}})} = \kappa \Delta T_{(f_{\mathcal{S}})} + f_{\mathcal{S}}, & \text{in } D_c^{(0)} \times [0, t_f] \\ \kappa \nabla T_{(f_{\mathcal{S}})} \cdot \mathbf{n}_S = q(t), & \text{on } S \times [0, t_f] \\ \kappa \nabla T_{(f_{\mathcal{S}})} \cdot \mathbf{n} = 0, & \text{on } \partial D_c^{(0)} / S \times [0, t_f] \\ T_{(f_{\mathcal{S}})}(0, \cdot) = T_0(\cdot), & \text{in } D_c^{(0)}. \end{cases} \quad (\text{o.1})$$

it holds that $T_{(f_{\mathcal{S}})}(t, x, y) = T^{(\mathcal{S})}(t, x, y)$ for all $(x, y) \in S$ and for all $t \in [0, t_f]$ and $f_{\mathcal{S}}$ has compact support within the region corresponding to the void.

Then, our inverse problem becomes to estimate the fictitious heat source $f_{\mathcal{S}}(t, x, y)$ in (o.1).

¹Giusteri GG, Marcuzzi F, Rinaldi L. Replacing voids and localized parameter changes with fictitious forcing terms in boundary-value problems, Results in Applied Mathematics, **20**, 2023, 10.1016/j.rinam.2023.100402

In¹ the source term estimator f_{mean}^k be derived from the PDE equation as

$$f_{mean}^k = \frac{1}{N_t} \sum_t \left(\rho C_t \mathbf{e}_{\beta k} + \kappa_x^2 \mathbf{e}_{\beta k} + \kappa_y^2 \mathbf{e}_{\beta k} \right), \quad (o.2)$$

where derivatives are approximated with a suitable finite-difference scheme using the measured values and we assume to accept the approximation $\frac{\partial}{\partial y} \mathbf{e}_{\beta k} \approx 0$ because it cannot be computed directly.

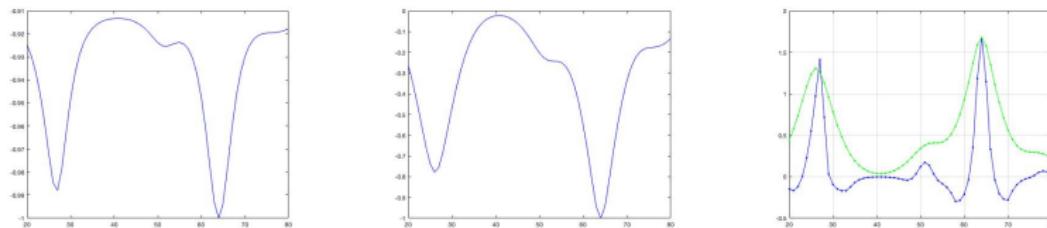


Figure: On the left and on the center, respectively, the prediction error with opposite sign $-\mathbf{e}_{\beta k}$ at the first iteration of the sparse test problem, whose pattern is shown on the left, with a lower and a bigger value of the conductivity coefficient κ . On the right the same prediction error $\mathbf{e}_{\beta k}$ with the higher value of κ (light line) together with the external source estimate obtained with (o.2) (dark line).

¹Dessole M., Marcuzzi F., Accurate detection of hidden material changes as fictitious heat sources, Numerical Heat Transfer, Part B: Fundamentals, 2023, doi 10.1080/10407790.2023.2220905

The fictitious heat source can be estimated from a dynamical system in the so-called *state-space* form:

$$x(k+1) = A \cdot x(k) + B \cdot u(k) + v(k) \quad (o.3)$$

$$y(k) = C \cdot x(k) + D \cdot u(k) + w(k) \quad (o.4)$$

where $\{x(k)\}$ is the state vector, $\{y(k)\}$ is the vector of measured outputs, $\{u(k)\}$ is the vector of inputs (which are supposed known), $\{v(k)\}$ and $\{w(k)\}$ are *model noise* and *measurement noise*, supposed gaussian, with zero mean and covariance matrices Q and R .

Then, under certain assumptions, it is possible to apply the Kalman Filter to estimate the state trajectory of the real system. Let us recall here the KF in its one-step version:

$$P(k) = \left[\left(Q(k-1) + A'(k-1) P(k-1) A'(k-1)^T \right)^{-1} + C^T R^{-1} C \right]^{-1} \quad (o.5)$$

$$\delta\hat{x}(k) = -P(k) C^T R^{-1} [C(A'(k-1) \hat{x}(k-1) + B u(k-1)) - \bar{y}(k)] \quad (o.6)$$

$$\hat{x}(k) = A'(k-1) \hat{x}(k-1) + B u(k-1) + \delta\hat{x}(k) \quad (o.7)$$

Here the scientific machine learning can learn the covariance matrices and avoid to compute the Kalman gain¹.

¹Guy Revach, Nir Shlezinger, Xiaoyong Ni, A.L. Escoriza, Ruud J. G. Van Sloun and Yonina C. Eldar. KalmanNet: Neural Network Aided Kalman Filtering for Partially Known Dynamics, IEEE Transactions on Signal Processing, 70, pp.1532-1547, 2022

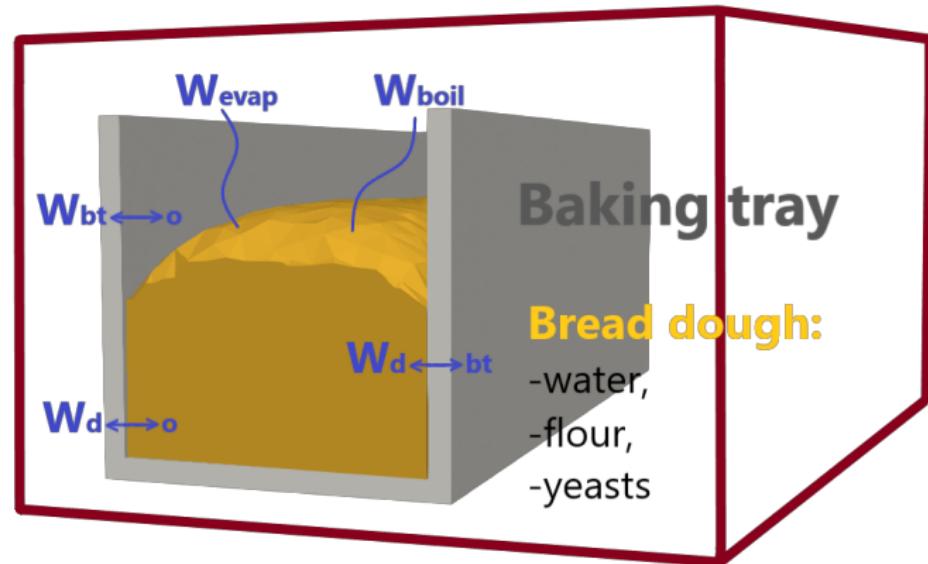
Sketch of the model: energy exchanges

iNEST

Fabio
Marcuzzi

A finite-element computational model for bread leavening and cooking will be presented. The partial differential equations that describe heat exchange and the presence of moisture, yeast and carbon dioxide are coupled with the quasi-static evolution of the growing elastic dough.

Hot oven



Elasticity with volumetric growth

iNEST

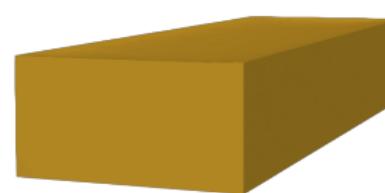
Fabio
Marcuzzi

The dough is treated as a hyperelastic material so, under the action of a body force B , it has an elastic behavior with energy:

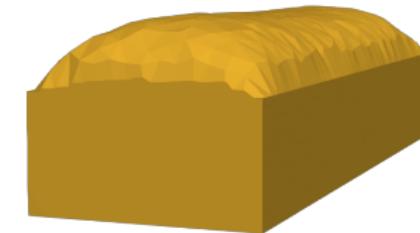
$$E(u) = \int_{\Omega} \frac{1}{2} (\operatorname{tr} C - 3 \sqrt[3]{J_{ref}^2}) + \frac{\lambda}{2} (\log(J) - \log(J_{ref}))^2 dX - \int_{\Omega} B \cdot u dX \quad (o.8)$$

where $F = \nabla_X u$, $C = F'F$ and $J = \det F$.

$$\text{Elasticity equation: } \delta E(u) = 0 \quad (o.9)$$



$$J_{ref} = 1$$



$$J_{ref} > 1$$

Heat equation in the bread

iNEST

Fabio
Marcuzzi

The temperature of the dough θ evolves according to the heat equation written in material coordinates:

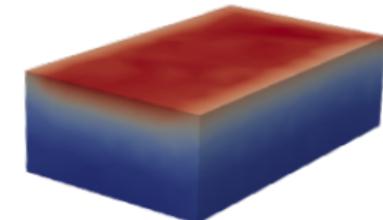
$$\int_{\Omega} \rho c_p J \frac{\partial}{t} \tilde{\theta} dX + \int_{\Omega} (k J C^{-1} \nabla_X \theta) \cdot (\nabla_X \tilde{\theta}) dX = \\ = W_{o \rightarrow d} + W_{bt \rightarrow d} + W_{loss} \quad (0.10)$$

where $\tilde{\theta}$ is the test function and:

$$W_{o \rightarrow d} = \int_{\Gamma_1} b_{od} (\theta_{out} - \theta) \tilde{\theta} ds$$

$$W_{bt \rightarrow d} = \int_{\Gamma_2} b_{bd} (\theta_{bt} - \theta) \tilde{\theta} ds$$

$$W_{loss} = \begin{cases} W_{evap} = - \int_{\Gamma_1} c_{cla} k_2 (\rho_m - \rho_{out}) \tilde{\theta} ds & \text{if } \theta < 100^{\circ}C \\ W_{boil} = - \int_{\Gamma_1} c_p k_3 (\theta_{out} - 100) \rho_m \tilde{\theta} ds & \text{if } \theta \geq 100^{\circ}C \end{cases}$$



Heat equation in the baking tray

iNEST

Fabio
Marcuzzi

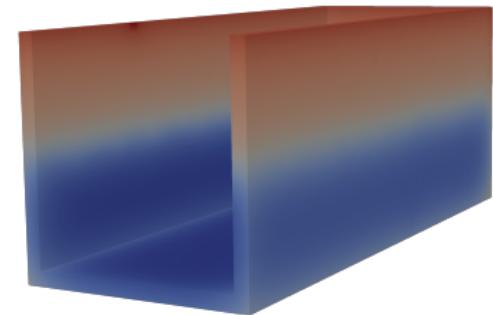
The temperature of the baking tray θ_{bt} changes according to the following heat equation:

$$\int_{\Omega_{bt}} \rho_{bt} c_{bt} \frac{\theta_{bt}}{t} \tilde{\theta}_{bt} dX + \int_{\Omega_{bt}} (k_{bt} \nabla_X \theta_{bt}) \cdot (\nabla_X \tilde{\theta}_{bt}) dX = \\ = W_{o \rightarrow bt} + W_{d \rightarrow bt} \quad (o.ii)$$

where $\tilde{\theta}_{bt}$ is the test function and:

$$W_{o \rightarrow bt} = \int_{\Gamma_{bt_1}} b_{ob} (\theta_{out} - \theta_{bt}) \tilde{\theta}_{bt} ds$$

$$W_{d \rightarrow bt} = \int_{\Gamma_{bt_2}} b_{db} (\theta - \theta_{bt}) \tilde{\theta}_{bt} ds$$





Moisture diffusion equation

iNEST

Fabio
Marcuzzi

The water density with respect to the volume of the dough ρ_m , has a diffusive evolution as follows:

$$\int_{\Omega} J \frac{\rho_m}{t} \tilde{\rho}_m dX + \int_{\Omega} (\beta J C^{-1} \nabla_X \rho_m) \cdot (\nabla_X \tilde{\rho}_m) dX = M_{loss} \quad (0.12)$$

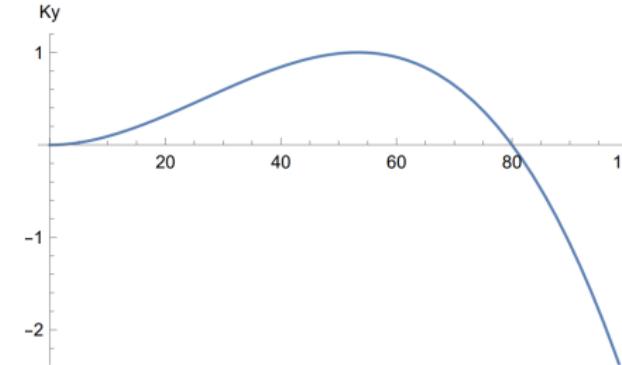
where $\tilde{\rho}_m$ is the test function and:

$$M_{loss} = \begin{cases} M_{evap} = - \int_{\Gamma_1} k_2 (\rho_m - \rho_{out}) \tilde{\rho}_m ds & \text{if } \theta < 100 {}^\circ C \\ M_{boil} = - \int_{\Gamma_1} k_3 (\theta_{out} - 100) \rho_m \tilde{\rho}_m ds & \text{if } \theta \geq 100 {}^\circ C \end{cases}$$

Rate equation for the yeast evolution

It has been supposed that the yeasts are uniformly distributed in the dough.

Their concentration with respect to the flour quantity Y evolves in time, according to the temperature θ :



$$\begin{cases} \frac{d}{dt} Y = K_y(\theta) Y & \implies Y(t) = \exp(K_y(\theta) t) Y_0 \\ Y(0) = Y_0 \end{cases} \quad (0.13)$$

Y_0 is the initial concentration and the growth rate is:

$$K_y(\theta) = \gamma_{cost} \frac{(80 - \theta) 27}{80^3} \frac{\theta^2}{4}.$$

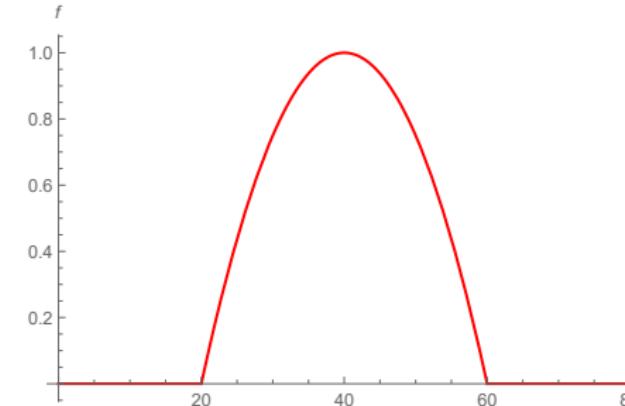
CO_2 production and diffusion

iNEST

Fabio

Marcuzzi

The metabolism of the yeasts implies a CO_2 production, so there is an evolution of its concentration D :



$$\int_{\Omega} J \frac{D}{t} \tilde{D} dX + \int_{\Omega} (\beta_{co2} J \mathbf{C}^{-1} \nabla_X D) \cdot (\nabla_X \tilde{D}) dX = \int_{\Omega} f(\theta) Y \tilde{D} dX \quad (o.14)$$

where \tilde{D} is the test function and the CO_2 production rate is:

$$f(\theta) = \max \left(-D_{co2} \frac{(\theta - 20)(\theta - 60)}{400}, 0 \right).$$



Volume expansion

iNEST

Fabio
Marcuzzi

The rate of CO_2 involves a volume expansion because of the leavening. From the ideal gas law:

$$Vol_D = \frac{\theta n_{mol} R}{P_a} \quad (o.15)$$

$$J_{ref} = 1 + \frac{Vol_D D}{Q} \quad (o.16)$$

where Vol_D and Q are the volume of one gram of CO_2 and of one gram of dough respectively.

Also the other elastic stiffness change according to the CO_2 quantity, so (at every 10 steps) the elasticity equation is rerun with these updated parameters in order to obtain the volume expansion due to the current CO_2 value.

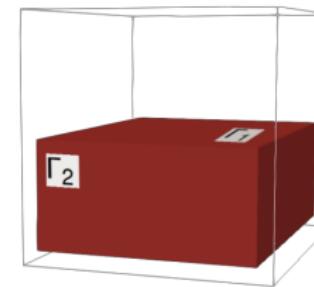
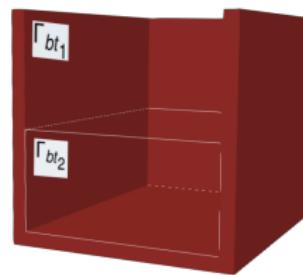
The meshes

iNEST

Fabio

Marcuzzi

In order to approach the numerical model we construct the geometries and define the meshes on them by choosing the resolution. We divided the boundary of the baking tray and of the dough to manage the heat flux between the two.



Γ_2 and Γ_{bt_2} are the parts of boundary of bread in contact with the baking tray. Γ_1 and Γ_{bt_1} are in contact with the air in the oven.

Since we use material coordinates the domains are fixed, independently of the deformations.



The spaces of finite elements

iNEST

Fabio
Marcuzzi

The spaces of finite elements are defined to the end of approximating functions with polynomials:

Space of functions	Functions approximated
V_b is a Vector Function Space defined on meshbr of degree-1 polynomials.	Placement \boldsymbol{u} .
W_b is a Function Space defined on meshbr of degree-2 polynomials.	Temperature of the dough θ , density ρ , moisture yeasts rate Y and CO_2 concentration D .
U_b is a Function Space defined on meshbt of degree-1 polynomials.	Temperature of the baking tray θ_{bt} .



Time evolution

iNEST

Fabio
Marcuzzi

A semi-implicit Euler method is used to deal with the coupling among the different equations

for n in range(n_{steps}):

- solve the heat eq. in the **bread dough**
- solve the diffusion eq. for the **moisture**
- solve the heat eq. in the **baking tray**
- compute the **yeasts** growth rate
- solve the diffusion eq. for the **CO_2 concentration**
- compute the volume occupied by CO_2

if $remainder(n + 1, 10) = 0$:

- update the value of J_{ref} and the elastic stiffness
 - solve the **elasticity** eq. (iterative nonlinear solver).
- end if**

end for

The simulations

iNEST

Fabio
Marcuzzi

The implementations using the Python library FEniCS give us the following simulations:

**The volume
expansion**



The partial differential equations that describe heat exchange are coupled with the quasi-static evolution of the growing elastic dough.

The baking tray temperature evolution



The bread temperature evolution



**The baking
process**

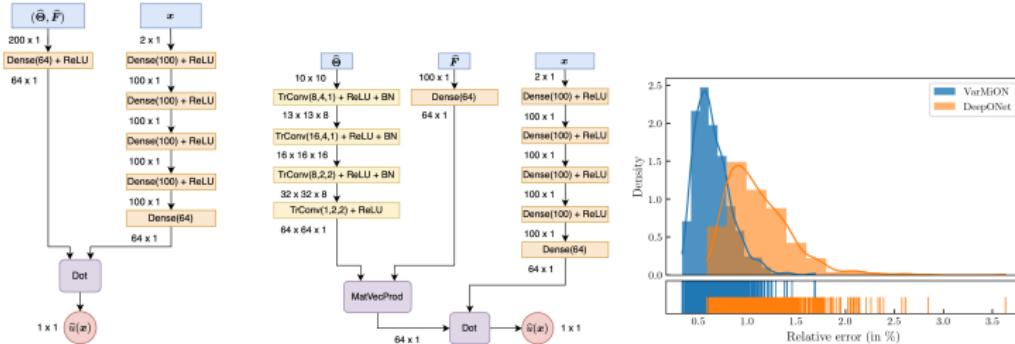


Physics-aware soft-sensor IV: inner voids/material changes estimation from thermographic inspection

iNEST

Fabio
Marcuzzi

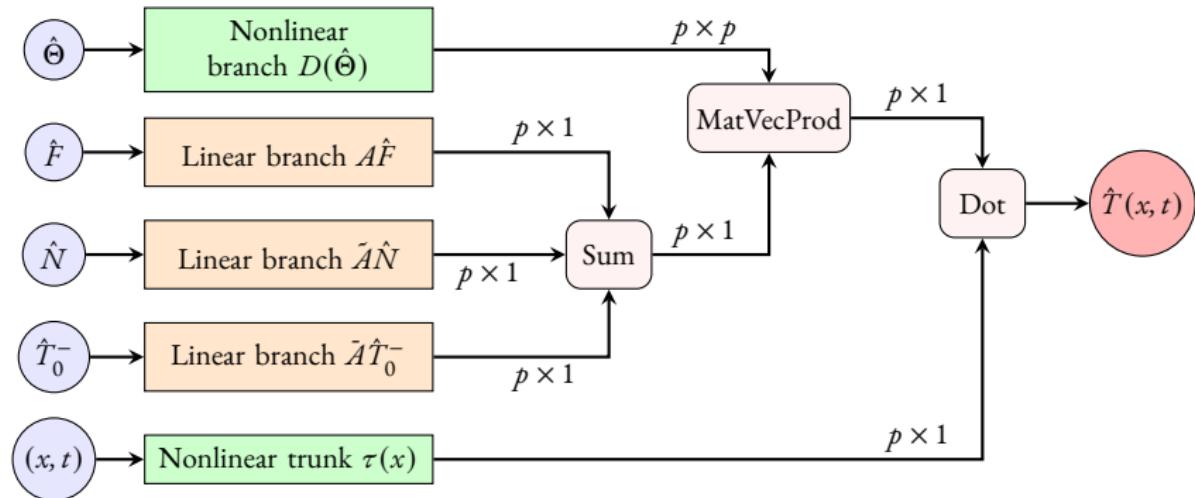
Surrogate modelling by scientific machine learning: the most interpretable method gets best results in force of an architecture that is more tight to the weak form used in Finite Element models.



An ARM4 microcontroller (144 MHz, with a HW single-precision floating-point unit) with its \sim 8MFLOPS, can compute the predicted temperature in a single-point in space in 0.09 seconds.

Physics-aware soft-sensor IV: inner voids/material changes estimation from thermographic inspection

iNEST

Fabio
Marcuzzi

Deep Kalman Filter

iNEST

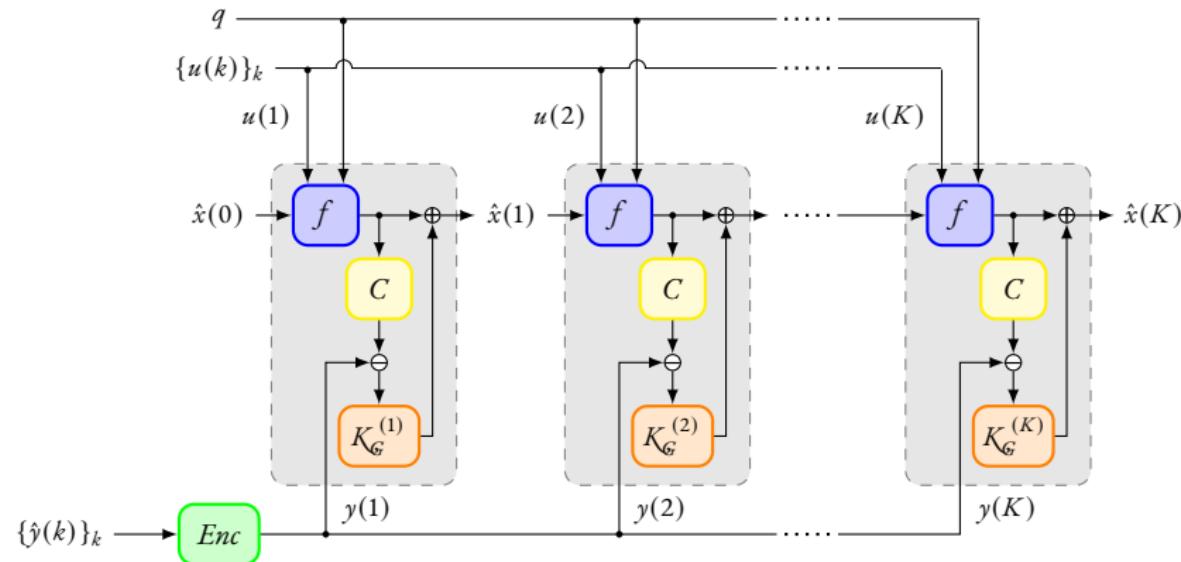
Fabio
Marcuzzi

Figure: Structure of the Deep Kalman Filter network.

²E. Chinellato, F. Marcuzzi, The Deep Kalman Filter, Springer LNCS 14836, Computational Science – ICCS 2024, Part V, Chapter 22 (2024) 307–321 doi:<https://doi.org/10.1007/978-3-031-63775-9>.

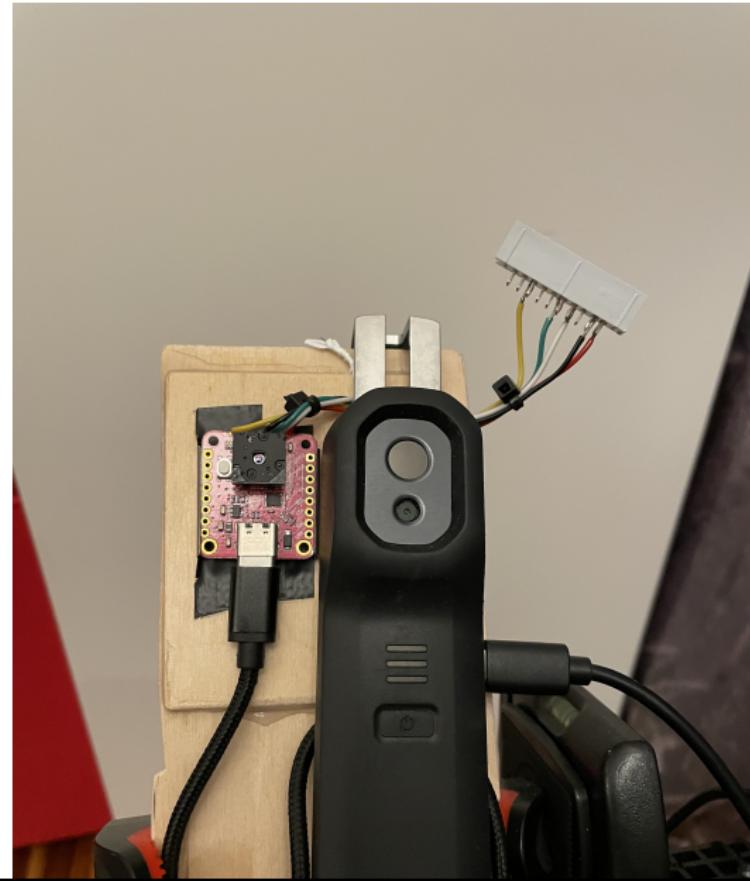
³E. Chinellato, Fabio Marcuzzi. State, parameters and hidden dynamics estimation with the Deep Kalman Filter: regularization strategies, submitted, 2024.

Physics-aware soft-sensor IV: inner voids/material changes estimation from thermographic inspection

iNEST

Fabio

Marcuzzi





iNEST

Fabio
Marcuzzi

Thank You for your attention!

questions? → marcuzzi@math.unipd.it

Source code will be available at the NLALDlab repository:
<https://github.com/NLALDlab>.