REFERENCES

[1] A. Bejan, *Advanced Engineering Thermodynamics*, 2nd ed., Wiley, New York, 1997.

[2] A. Bejan, *Shape and Structure from Engineering to Nature*, Cambridge University Press, Cambridge, UK, 2000.

[3] A. Bejan and E. Sciubba, The optimal spacing of parallel plates cooled by forced convection, *International Journal of Heat and Mass Transfer,* Vol. 35, No. 12, pp. 3259-3264, 1992.

[4] S. Mereu, E. Sciubba and A. Bejan, The optimal cooliing of a stack of heat generating boards with fixed pressure drop, flow rate and pumping power, *International Journal of Heat and Mass Transfer*, Vol. 36, pp.3677–3686, 1993.

[5] A. Bejan, A.J Fowler and G. Stanecu, The optimal spacing of between horizontal cylinders in a fixed volume cooled by Natural Convection, *International Journal of Heat and Mass Transfer*, Vol. 38, pp.2047–2055, 1995.

[6] G. Ledezema and A. Bejan, Optimal geometric arrangement of staggered vertical plates in natural convection, *International Journal of Heat and Mass Transfer*, Vol. 119, pp.700–708, 1997.

[7] A.J Fowler, G.A. Ledezema and A. Bejan, Optimal geometric arrangement of staggered plates in forced convection, *International Journal of Heat and Mass Transfer*, Vol. 40, pp.1795–1805, 1997.

[8] G. Stanecu, A.J. Fowler and A. Bejan, The optimal spacing of cylinders in a free-stream cross-flow forced convection, *International Journal of Heat and Mass Transfer*, Vol. 39, pp. 311–317, 1996.

[9] A. Bejan, The optimal spacing for cylinders in cross-flow forced convection, *International Journal of Heat and Mass Transfer*, Vol. 117, pp.767–770, 1995.

[10] G. Ledezema A.M. Morega and A. Bejan, Optimal spacing between pin fins with impinging flow, *Journal of Heat and Mass Transfer*, Vol. 118, pp.570–577, 1996.

[11] A. Bejan, Constructal-theory network of conducting paths for cooling a heat generating volume, *International Journal of Heat and Mass Transfer*, Vol. 40, pp. 799–816, 1997.

[12] A. Bejan and S. Lorente, *Design with Constructal Theory*, Wiley, Hoboken, 2008.

[13] L. Ghodoossi, Conceptual study on Constructal theory, *Energy. Conversion. Management*, Vol.45, pp. 1379–1395, 2004.

[14] S. Lee, Optimum design and selection of heat sinks, *IEEE Transactions on Components, Packaging and Manufacturing Technology – Part A*, Vol. 18, No. 4, pp. 812 – 817, 1995.

[15] H-S Kou, J-J Lee, and C-W Chen, Optimum thermal performance of microchannel heat sink by adjusting channel width and height, *International Communications in Heat and Mass Transfer* Vol. pp. 577–582, 2008.

[16] R. W. Knight, J. S. Goodling, and D. J. Hall, Optimal thermal design of Forced convection heat sinks - Analytical, *Journal of Electron Packaging* Vol. 113, pp. 313 – 321, 1991.

[17] Y.S. Muzychka, Constructal design of forced convection cooled microchannel heat sinks and heat exchangers, *International Journal of Heat and Mass Transfer*, Vol. 48, pp. 3119–3127, 2005.

[18] Y.S. Muzychka, Constructal multi-scale design of compact micro-tube heat sinks and heat exchangers, *International Journal of Thermal Science*. Vol. 46, pp. 245–252, 2007.

[19] T. Bello-Ochende, L. Liebenberg, J.P. Meyer, A.G. Malan and A. Bejan, Consrtuctal conjugate heat transfer in three-dimensional cooling channels, *International Journal of Heat and Mass Transfer*, Vol. 14, pp. 279 – 293, 2007.

[20] K.K. Ambatipidi and M.M. Rahman, Analysis of conjugate heat transfer in mircochannel heat sinks, *Numerical Heat Transfer, Part A: Applications*, Vol. 37, No. 7, pp. 711 – 731, 2000.

[21] L.G. Page, T. Bello-Ochende, and J.P. Meyer, Maximum heat transfer density rate enhancement from cylinders rotating in natural convection, *International Communications in Heat and Mass Transfer*, Vol. 38, pp. 1354–1359, 2011.

[22] B. Xu, K.T. Ooti, N.T. Wong and W.K. Choi, Experimental investigation of flow friction for liquid flow in micro-channels, *International Communications in Heat and Mass Transfer*, Vol. 27, No. 8, pp. 1165 – 1176, 200.

[23] X.F. Peng and G.P. Peterson, Convective heat transfer and flow friction for water flow in micro-channel structures, *International Journal of Heat and Mass Transfer*, Vol. 39, pp. 2599 – 2608, 1996.

[24] P. Lee, S.V. Garimella and D. Liu, Investigation of heat transfer in rectangular microchannels, *International Journal of Heat and Mass Transfer*, Vol. 48, pp. 1688-1704, 2005.

[25] T.M. Harms, M.J. Kazmierczak, and F.M. Gerner, Developing convective heat transfer in deep rectangular microchannels, *International Journal of Heat and Fluid Flow*, Vol. 20, pp. 149 – 157, 1999.

[26] J. Judy, D. Maynes, and B.W. Webb, Characterization of frictional pressure drop for liquid flows through microchannels, *International Journal of Heat and Mass Transfer*, Vol. 45, pp. 3477 – 3489, 2002.

[27] G. L. Morini, Single-phase convective heat transfer in microchannels: a review of experimental results, *International Journal of Thermal Sciences*, Vol. 43, pp. 631– 651, 2004.

[28] B.A. Jubran, M.A. Hamdan and R.M. Abdullah, Enhanced heat transfer, missing pin, and optimisation for cylindrical pin fin arrays, *ASME Journal of Heat Transfer*, Vol. 115, pp. 576 – 583, 1993.

[29] M. Tahat, Z.H. Kodah, B.A. Jarrah and S.D. Probert, Heat transfers from pin-fin arrays experiencing forced convection, *Applied Energy*, Vol. 67, pp. 419 – 442, 2000.

[30] G.A. Ledezma, A. Bejan, and M. R. Errera, Constructal Tree Networks for Heat Transfer, *Journal of Applied Physics*, Vol. 82, No. 1, 1997, pp. 89 – 100.

[31] A. Bejan, G. Tsatsaronis and M. Moran, *Thermal design and optimisation*, New York: John Wiley. 1996.

[32] J.C. Ordonez and A. Bejan, Entropy generation minimization in parallel- plate counterflow heat exchangers. *International Journal of Energy Research*, Vol. 24, pp. 843 – 864, 2000.

[33] A. Bejan, S. Lorente and J. Lee, Unifying constructal theory of tree roots, canopies and forests, *Journal of Theoretical. Biology*, Vol. 254, pp. 529–540, 2008.

[34] T. Bello-Ochende and A. Bejan, Fitting the duct to the ‘‘body” of the convective flow, *International Journal of Heat and Mass Transfer*, Vol. 46, pp. 1693–1701, 2006.

[35] H. Wang, W. Dai and A. Bejan, Optimal temperature distribution in a 3D triple-layered skin structure embedded with artery and vein vasculature and induced by electromagnetic radiation, *International Journal of Heat and Mass Transfer*, Vol. 50, pp. 1843 – 1854, 2007.

[36] A. Bejan, V. Badescu and A. De Vos, Constructal theory of economics structure generation in space and time, *Energy Conversion Management*. Vol. 41, pp. 1429 –1451, 2000.

[37] A. Bejan, V. Badescu, and A. De Vos, Constructal theory of economics, *Applied Energy*. Vol. 67, pp. 37 – 60, 2000.

[38] A. Bejan, Optimal internal structure of volumes cooled by single phase forced and natural convection, *Journal of Electronic Packaging*, Vol. 125, 200–207, 2003.

[39] A. Bejan, Constructal theory of pattern formation. *Hydrology and Earth System Science*, Vol. 11, pp. 753- 768, 2007.

[40] A. Bejan, Why university rankings do not change: education as a natural hierarchical flow architecture. *International Journal of Design and Nature and Ecodynamics*, Vol. 2, pp. 319–27, 2007.

[41] A. Bejan, Two hierarchies in science: the free flow of ideas and the academy, *International Journal of Design and Nature and Ecodynamics*, Vol. 4, pp. 386–94, 2009.

[42] G. Weinerth, The constructal analysis of warfare. *International Journal of Design and Nature and Ecodynamics*, Vol. 5, pp. 268–76, 2010.

[43] A. Bejan, and J.H. Marden, Unifying constructal theory for scale effects in running, swimming and flying. *Journal of Experimental Biology*, Vol. 209, pp. 238–48, 2008.

[44] J. D Charles, and A. Bejan, The evolution of speed, size and shape in modern athletics, *Journal of Experimental Biology* Vol. 2012, pp. 2419 – 25, 2009.

[45] A. Bejan, E.C. Jones and J.D Charles. The evolution of speed in athletics: why the fastest runners are black and swimmers white, *International Journal of Design and Nature and Ecodynamics*, Vol. 5, pp. 199–211, 2010.

[46] A. H. Reis, A.F Miguel and M. Aydin. Constructal theory of flow architecture of the lungs. *Journal of Medical Physics*, Vol. 31, pp. 1135–40, 2004.

[47] A.F Miguel, Constructal pattern formation in stony corals, bacterial colonies and plant roots under different hydrodynamics conditions, *Theoretical Biology*, Vol. 242, pp. 954–61, 2006.

[48] A. Nakayama, F. Kuwahara and W. Liu, A macroscopic model for counter-current bioheat transfer in a circulatory system. *Journal of Porous Medium*, Vol. 12, pp. 289–300, 2009.

[49] S. Quéré, Constructal theory of plate tectonics, *International Journal of Design and Nature and Ecodynamics*, Vol. 5 pp. 242–53, 2010.

[50] A.H Reis, C. Gama, Sand size versus beachface slope – an explanation based on the constructal law. *Geomorphology*, Vol. 114, pp. 276–83, 2010.

[51] J.P., Meyer, Constructal law in technology, thermofluid and energy systems, and in design education, *Physics of Life Reviews*, Vol. 8, pp. 247 – 248, 2011.

[52] A. Bejan and S. Lorente, The constructal law and the evolution of design in nature, *Physics of Life Reviews*, Vol. 8, pp. 209 – 240, 2011.

[53] T. Basak, The law of life: the bridge between physics and biology, *Physics of Life Reviews*, Vol. 8, pp. 249 – 252, 2011.

[54] A.F. Miguel, The physics principle of generation of flow configuration, *Physics of Life Reviews*, Vol. 8, pp. 243 – 244, 2011.

[55] L. A. O. Rocha, Constructal law: from law of physics to applications and conferences, *Physics of Life Reviews*, Vol. 8, pp. 245–246, 2011.

[56] G. Lorenzini and C. Biserni, The constructal law: from design in nature to social dynamics and wealth as physics, *Physics of Life Reviews*, Vol. 8, pp. 259–260, 2011.

[57] J.O. Tuhtan, Go with the flow: connecting energy demand, hydropower, and fish with constructal theory, *Physics of Life Reviews*, Vol. 8, pp. 253–254, 2011.

[58] L. Wang, Universality of design and its evolution, *Physics of Life Reviews*, Vol. 8, pp. 257-258, 2011.

[59] Y. Ventikos, The importance of the constructal framework in understanding and eventually replicating structure in tissue, *Physics of Life Reviews*, Vol. 8, pp. 241–242, 2011.

[60] Y. S. Kim, Design with Constructal Theory: Steam Generators, Turbines and Heat Exchangers, PhD thesis, Duke University, USA, 2010.

[61] S. Kim, Constructal vascular composites for cooling and heating, PhD thesis, Duke University USA, 2008.

[62] A. K. Pramanick, Natural Philosophy of Thermodynamic Optimization, PhD thesis Duke University Indian Institute of Technology Kharagpur, 2007.

[63] R. S. Kulkarni, Infusion of Robustness into the Product Platform Constructal Theory Method, MSc thesis, Georgia Institute of Technology, 2005.

[64] E. Cetkin, The Natural Emergence of Vascular Design with Turbulent Flow, MSc, thesis, Duke University, USA, 2010.

[65] L.A.O. Rocha, E. Lorenzini and C. Biserni, Geometric optimization of shapes on the basis of Bejan's Constructal theory, *International Communications in Heat and Mass Transfer*, Vol 32, pp. 1281–1288, 2005.

[66] T. Bello-Ochende, J. P. Meyer and A. Bejan, Constructal multi-scale pin fins, *International Journal of Heat and Mass Transfer*, Vol. 53, pp. 2773–2779, 2010.

[67] T. Bello-Ochende, J.P. Meyer and O.I. Ogunronbi, Constructal multiscale cylinders rotating in cross-flow, *International Journal of Heat and Mass Transfer*, Vol. 54, pp.2568–2577, 2011.

[68] A. Bejan and Y. Fautrelle, Constructal multi-scale structure for maximal heat transfer density, *Acta Mechanica*, Vol. 163, pp. 39–49, 2003.

[69] A. Bejan , L.A.O. Rocha and S. Lorente, Thermodynamic optimization of geometry: T- and Y-shaped constructs of fluid streams , *International Journal of Thermal Sciences*, Vol. 39, pp. 949–960, 2000.

[70] A. Bejan, and N. Dan, Two constructal routes to minimal heat flow resistance via greater internal complexity, *Journal of Heat Transfer*, Vol. 121, pp. 6 – 14, 1999.

[71] L.A.O. Rocha, S. Lorente, and A. Bejan, Constructal design for cooling a disc-shaped area by conduction, *International Journal of Heat and Mass Transfer*, Vol. 45, pp. 1642 – 1652, 2003.

[72] A.H., Reis, Constructal Theory – Complex flow structures in engineering and in Nature III Conferência Nacional em Mecânica de Fluidos, Termodinâmica e Energia (MEFTE - BRAGANÇA 09), pp. 1– 17, 2009.

[73] A. H. Reis, Constructal Theory: From Engineering to Physics, and How Flow Systems Develop Shape and Structure, *Applied Mechanics Reviews*, Vol. 59, pp. 269–282, 2006.

[74] Y. Fan, and L. Luo, Recent Applications of Advances in Microchannel Heat Exchangers and Multi-Scale Design Optimisation, *Heat Transfer Engineering*, Vol. 29, pp. 461–474, 2008.

[75] H. H. Bau, optimisation of conduits' shape in micro heat exchangers, *International Journal of Heat and Mass Transfer*, Vol. 43, pp. 2717–2723, 1998.

[76] A. Yilmaz, O. Buyukalaca and T. Yilmaz, Optimum shape and dimensions of ducts for convective heat transfer in laminar flow at constant wall temperature, *International Journal of Heat and Mass Transfer*, Vol. 43, pp. 767–775, 2000.

[77] W Yang, T. Furukawa and S. Torii, Optimal package of stacks of convection – Cooled printed circuit boards using entropy generation minimization method, *International Journal of Heat and Mass Transfer* Vol. 51, pp. 4038 – 4046, 2008.

[78] A. Bejan, and Y. Fautrelle, Constructal multi-scale structure for maximal heat transfer density, *Acta Mechanical.* Vol. 163, pp. 39–49, 2003.

[79] R. W. Knight, J. S. Goodling, and D. J. Hall, Optimal thermal design of Forced convection heat sinks - Analytical, *Journal of Electronic Packaging*, Vol. 113, pp. 313 – 321, 1991.

[80] R. W. Knight, D. J. Hall, J. S. Goodling, and R. C. Jaeger, Heat sink optimization with application to microchannels, *IEEE Transaction on Component, Hybrids, Manufacturing. Technology.* Vol. 15, pp. 832–842, 1992.

[81] R. W. Knight, J. S. Goodling, and B. E. Gross "Optimal thermal design of air cooled forced convection finned heat sinks – e*xperimental verification, – Intersociety Conference on Thermal Phenomena*, 1992.

[82] D.C. Knupp, C.P. Naveira-Cotta and R.M. Cotta, Theoretical analysis of conjugated heat transfer with a single domain formulation and integral transforms, *International Communications in Heat and Mass Transfer*, Vol. 39, pp. 355–362, 2012.

[83] S.H. Kim and N.K. Anand, Laminar developing flow and heat transfer between a series of parallel plates with surface mounted discrete heat sources, *International Journal of Heat and Mass Transfer,* Vol. 37, pp. 2231–2244, 1994.

[84] J.H. Ryu, D.H. Choi, and S.J. Kim, Numerical optimization of the thermal performance of a microchannel heat sinks, *International Journal of Heat and Mass Transfer*, Vol. 45, pp. 2823 – 2827, 2002.

[85] A. G. Fedorov and Viskanta, Three-dimensional conjugate heat transfer in the microchannel heat sink for electronic packaging, *International Journal of Heat and Mass Transfer*, Vol.43, pp. 399 – 415, 2000.

[86] W. Qu, and I. Mudawar, Analysis of three-dimensional heat transfer in micro-channel heat sinks, *International Journal of Heat and Mass Transfer,* Vol. 45, pp. 3973 – 3985, 2002.

[87] R.K. Shah and A.L. London, *Laminar Flow Forced Convection in Ducts: a source Book for Compact Heat Exchanger Analytical Data, Supplement. 1*, Academic Press, New York, 1978.

[88] K. Kawano, K. Minakami, H. Iwasaki and M. Ishizuka, Microchannel Heat Exchanger For Cooling Electrical Equipment, *Application of Heat Transfer In*

Equipment, Systems And Education, ASME HTD-361-3/PID- 3, pp. 173–180, 1998.

[89] K.K. Ambatipudi, and M.M. Rahman, Analysis of conjugate heat transfer in microchannel heat sinks, *Numerical Heat Transfer* Part A Vol. 37, pp. 711–731, 2000.

[90] C. Chen, Forced convection heat transfer in microchannel heat sinks, *International Journal of Heat and Mass Transfer,* Vol. 50, pp. 2182–2189, 2007.

[91] A.K. da Silva, S. Lorente and A. Bejan, Optimal distribution of discrete heat sources on a plate with laminar forced convection, *International Journal of Heat and Mass Transfer,* Vol. 47, pp. 2139–2148, 2004.

[92] A.K. da Silva, S. Lorente and A. Bejan, Optimal distribution of discrete heat sources on a wall with natural convection, *International Journal of Heat and Mass Transfer*, Vol. 47 pp. 203–214, 2004.

[93] T. Bello-Ochende and A. Bejan, Maximum heat transfer density: Plates with multiple lengths in forced convection *International Journal Thermal Sciences* Vol. 43, pp. 1181–1186, 2004.

[94] T. Bello-Ochende, L. Liebenberg and J.P. Meyer, Constructal cooling Channels for micro-channel heat sinks, *International Journal of Heat and Mass Transfer*, Vol. 50, pp. 4141 – 4150, 2007.

[95] T. Bello-Ochende, L. Liebenberg and J. P. Meyer, Constructal Design: Geometric optimization of micro-channel heat sinks, *South African Journal of Science*, Vol. 103, 12, pp. 483 – 489, 2007.

[96] T. Bello-Ochende, L. Liebenberg, A. G. Malan, A. Bejan and J. P. Meyer, Optimization of conjugate heat transfer in three-dimensional cooling channels, *Journal of Enhanced Heat Transfer*, Vol. 14, pp. 279 – 293, 2007.

[97] T. Bello-Ochende T and J. P. Meyer, Constructal cooling channels: application to heat transfer in micro-channel heat sinks, *International Journal of Emerging Multidisciplinary Fluid Sciences*, Vol. 1, pp. 61 – 83, 2009.

[98] T. Bello-Ochende, J. P. Meyer and A. Bejan, Constructal ducts with wrinkled entrances, *International Journal of Heat Mass Transfer*, Vol. 52, pp. 3628 – 3633, 2009.

[99] T. Bello-Ochende, J. P. Meyer and F. U. Ighalo, Combined numerical and constructal theory for the design of micro-channel heat sinks, *Journal of Numerical Heat Transfer, Part A*: Applications, Vol. 58 pp. 882 – 899, 2010.

[100] T. Bello-Ochende and A. Bejan, Optimal spacing for mixed convection, *Journal of Heat Transfer*, Vol. 126, pp. 956 – 962, 2004.

[101] T. Bello-Ochende, A. Bejan, Constructal multi-scale cylinder in cross flow, International *Journal of Heat and Mass Transfer*, Vol. 48, pp. 1373 – 1383, 2005.

[102] T. Bello-Ochende, and A. Bejan, Constructal multi-scale cylinders with natural convention, *International Journal of Heat and Mass Transfer*, Vol. 48, pp. 4300 – 4306, 2005.

[103] T. Bello-Ochende, J. P. Meyer and J. Dirker, Three-dimensional Multi-Scale Plate Assembly for Maximum heat Transfer Rate Density, *International Journal of Heat Mass Transfer*, Vol. 53, pp. 586 – 593, 2010.

[104] M.R. Rajkumar, G. Venugopal and S. A. Lal, Numerical study of natural convection from a heat-generating element using a locally divergence free FEM and comparison with experiment, *International Communications in Heat and Mass Transfer* Vol. 39, pp. 530–536, 2012.

[105] H.A. Mohammed, P. Gunnasegaran and N.H. Shuaib, Influence of channel shape on the thermal and hydraulic performance of microchannel heat sink, *International Communications in Heat and Mass Transfer*, Vol. 38 pp. 474 – 480, 2011.

[106] L.A.O. Rocha, E. Lorenzini, C. Biserni and Y. Cho, Constructal design of a cavity cooled by convection, *International Journal of Design and Nature and Ecodynamics*, Vol. 5, pp. 212–220, 2010.

[107] C. Biserni, L.A.O. Rocha G. Stanescu and E. Lorenzini, Constructal H-shaped cavities according to Bejan's theory *International Journal of Heat and Mass Transfer*, Vol. 50, 2132–2138, 2007.

[108] C. Biserni, L.A.O. Rocha and A. Bejan, Inverted fins: geometric optimization of the intrusion into a conducting wall, *International Journal of Heat and Mass Transfer*, Vol. 47, pp. 2577–2586, 2004.

[109] G. Lorenzini, and L.A.O. Rocha, Geometric optimization of T-Y-shaped cavity according to constructal design, *International Journal of Heat Mass Transfer*, Vol. 52 pp. 4683–8, 2009.

[110] G. Lorenzini, C. Biserni, and L.A.O. Rocha, Geometric optimization of isothermal cavities according to Bejan's theory. *International Journal of Heat Mass Transfer*, Vol. 54 pp. 3868–73, 2011.

[111] G. Lorenzini, and L.A.O. Rocha, Constructal design of Y-shaped assembly of fins. *International Journal of Heat Mass Transfer*, Vol. 49, pp. 4552–7, 2006.

[112] G. Lorenzini, and L.A.O. Rocha, Constructal design of T-Y assembly of fins for an optimized heat removal, *International Journal of Heat Mass Transfer*, Vol. 52, pp. 458–63, 2008.

[113] G. Lorenzini, R.L. Corrêa, E.D. Dos Santos and L.A.O. Rocha, Constructal design of complex assembly of fins, *ASME Journal of Heat Transfer*, Vol. 133, pp. 071801-1 – 071801-10, 2011.

[114] M. R. Salimpour, M. Sharifhasan, and E. Shirani, Constructal optimization of the geometry of an array of micro-channels, *International Communication of Heat and Mass Transfer*, Vol. 38, pp. 93–99, 2011.

[115] R.S. Matos, J.V.C. Vargas and A. Bejan, Three-dimensional optimization of staggered finned circular and elliptic tubes in forced convection, *International Journal of Heat and Mass Transfer*, Vol. 44, pp. 3953–3961, 2000.

[116] R.S. Matos , T.A. Laursen, J.V.C. Vargas and A. Bejan, Three-dimensional optimization of staggered finned circular and elliptic tubes in forced convection, *International Journal of Thermal Sciences* Vol. 43, pp. 477–487, 2004.

[117] J.C. Ordonez, Integrative energy-systems design: system structure from thermodynamic optimization, PhD thesis, Duke University, USA, 2003.

[118] J. Dirker and J.P. Meyer, Cooling layers in rectangular heat-generating electronic regions for two boundary condition types: A comparison with a

traditional approach, *South African Journal of Science* Vol. 103, pp. 474 – 482, 2007.

[119] A.H. Reis, A.F. Miguel and A. Bejan, Constructal Theory of Particle Agglomeration of Design of Air-cleaning Devices, *Journal of Physics D: Applied Physics*, Vol. 39, pp. 3086–3096, 2006.

[120] J.V.C. Vargas, J.C. Ordonez and A. Bejan Constructal flow structure for a PEM fuel cell, *International Journal of Heat and Mass Transfer*, Vol. 47, pp. 4177–4193, 2004.

[121] J. C. Ordonez, S. Chen, J. V. C. Vargas, F. G. Dias, J. E. F. C. Gardolinski and D. Vlassov, Constructal flow structure for a single SOFC, *International Journal of Energy Research*, Vol. 31, pp. 1337–1357, 2007.

[122] H., Wen , J.C., Ordonez and J.V.C, Vargas, Single solid oxide fuel cell modeling and optimization, *Journal of Power Source*, Vol. 196, pp. 7519–7532, 2011.

[123] S. O. Obayopo, T. Bello-Ochende and J.P. Meyer, Three-dimensional optimisation of a fuel gas channel of a proton exchange membrane fuel cell for maximum current density, *international journal of energy research*, 2011 doi: 10.1002/er.1935.

[124] S. O. Obayopo, T. Bello-Ochende and J.P. Meyer, Modelling and optimization of reactant gas transport in a PEM fuel cell with a transverse pin fin insert in channel flow, *International journal of hydrogen energy* Vol. 37, pp. 10286–10298, 2012.

[125] S.R. White, N.R. Sottos, J. Moore, P. Geubelle, M. Kessler, E. Brown, S. Suresh and S. Viswanathan, Autonomic Healing of Polymer Composites, *Nature*, Vol. 409, pp. 794–797, 2001.

[126] A. Bejan, S. Lorente and K.-M. Wang, Network of Channels for Self-healing Composite Materials, *Journal of Applied Physics*, Vol. 100, pp. 033528–033528-6, 2006.

[127] K.-M. Wang, S. Lorente and A. Bejan, Vascularised Networks with Two Optimised Channels Sizes, *Journal of Physics D: Applied Physics*, Vol. 39, pp. 3086–3096, 2006.

[128] S.W. Kim, S. Lorente and A. Bejan, Vascularised Materials: Tree-shaped flow architectures matched canopy to canopy, *Journal of Applied Physics*, Vol. 100, pp. 063525–063525-8, 2006.

[129] S. Lorente and A. Bejan, Heterogeneous porous media as multiscale structures for maximum flow access, *Journal of Applied Physics*, Vol. 100, pp. 114909–114909-8, 2006.

[130] S.W. Kim, S. Lorente, and A. Bejan, 'Vascularised materials with heating from one side and coolant forced from the other side', *International Journal of Heat and Mass Transfer*, Vol. 50, pp. 3498–3506, 2007.

[131] K-H. Cho, J. Lee, H.S. Ahn, A. Bejan and M.H. Kim, Fluid Flow and Heat Transfer in Vascularised Cooling Plates, *International Journal of Heat and Mass Transfer*, Vol. 53, pp. 3607–3614, 2010.

[132] S. Lorente, and A. Bejan, Svelteness, freedom to morph, and constructal multi-scale flow structures, *International Journal of Thermal Sciences*, Vol. 44, pp. 1123–1130, 2005.

[133] K-M. Wang, S. Lorente and A. Bejan, Vascularisation with grids of channels: multiple scales, loops and body shapes, *Journal of Physics D, Applied Physics*, Vol. 40, pp. 4740–4749, 2007.

[134] S. Kim, S. Lorente and A. Bejan, Dendritic vascularisation for countering intense heating from the side, *International Journal of Heat and Mass Transfer* Vol. 5, pp. 5877–5886, 2008.

[135] S. Kim, S. Lorente, A. Bejan, W. Miller and J. Morse, The emergence of vascular design in three dimensions , *Journal of Applied Physics* Vol. 103, pp. 123511-1 123511-8, 2008.

[136] K-M. Wang, S. Lorente and A. Bejan, Vascular materials cooled with grids and radial channels, *International Journal of Heat and Mass Transfer*, Vol. 52, pp. 1230–1239, 2009.

[137] E. Cetkin, S. Lorente, and A. Bejan, Natural constructal emergence of vascular design with turbulent flow, *Journal of Applied Physics*, Vol. 107, pp. 114901-1–114901-9, 2010.

[138] A. V. Azad and M. Amidpour, Economic Optimization of Shell and Tube Heat Exchanger based on Constructal Theory, *Energy*, Vol. 36 pp. 1087 – 1096, 2011.

[139] A. Beyene, and J. Peffley, Constructal Theory, Adaptive Motion, and Their Theoretical Application to Low-speed Turbine Design, Journal of Energy Engineering -ASCE, Vol. 135, pp. 112 – 118, 2009.

[140] Y. Kim, S. Lorente, and A. Bejan, Constructal Multi-tube Configuration for Natural and Forced convection in cross-flow *International Journal of Heat Mass Transfer*, Vol. 53, pp. 5121 – 5128, 2010.

[141] Y. Kim, S. Lorente, and A. Bejan, Steam Generator structure: continuous model and constructal design, *International Journal of Energy Research*, Vol. 35, pp. 336 – 345, 2011.

[142] A.Y. Alharbi, D.V. Pence and R.N. Cullion, Fluid flow through microscale fractallike branching channel networks, *Journal of Fluids Engineering*, Vol. 125, 1051–1057, 2003.

[143] A.H. Reis, A.F. Miguel and M. Aydin, Constructal theory of flow architecture of lungs, *Journal of Medical Physics*. Vol. 31, pp. 1135–1140, 2004.

[144] C. Amador, A. Gavriilidis, P. Angeli, Flow distribution in different microreactor scale-out geometries and the effect of manufacturing tolerances and channel blockage. *Chemical Engineering Journal*, Vol. pp. 101, 379–90, 2004.

[145] Y. Azoumah, N. Mazet and P. Neveu, Constructal network for heat and mass transfer in a solid–gas reactive porous medium, *International Journal of Heat Mass Transfer*, Vol. 47, pp. 2961–2970, 2004.

[146] D. Tondeur and L. Luo, Design and scaling laws of ramified fluid distributors by the constructal approach, *Chemical Engineering Science*. Vol. 59, pp. 1799–1813, 2004.

[147] L. Luo and D. Tondeur, Multiscale optimization of flow distribution by Constructal approach, *China Particuology*, Vol. 3, pp. 329–336, 2005.

[148] A. Bhakta and S. Bandyopadhyay, Constructal optimization of top contact metallization of a photovoltaic solar cell. *International Journal of Thermodynamics*, Vol. 8, pp. 175–81, 2005.

[149] Y. Chen and P. Cheng, An experimental investigation on the thermal efficiency of fractal tree-like microchannel nets, *International Communications in Heat and Mass Transfer*, Vol. 32, pp. 931–938, 2005.

[150] A.H. Reis and A.F. Miguel, Constructal theory and flow architectures in living systems, *Journal of Thermal Science*, Vol. 10, 57–64, 2006.

[151] A.-H. Wang, X.-G. Liang and J.-X. Ren, Constructal enhancement of heat conduction with phase change, *International Journal of Thermophysics*, Vol. 27, pp. 126–138, 2006.

[152] A.H. Reis, Constructal theory: from engineering to physics, and how flow systems develop shape and structure, *Applied Mechanical Review*, Vol. 59, pp. 269–282, 2006.

[153] A.H. Reis, Constructal view of scaling laws of river basins, *Geomorphology* Vol. 78, pp. 201–206, 2006.

[154] A.F. Miguel, Constructal pattern formation in stony corals, bacterial colonies and plant roots under different hydrodynamics conditions, *Journal of Theoretical Biology*. Vol. 242, pp. 954–961, 2006.

[155] X.-Q. Wang, A.S. Mujumdar and C. Yap, Numerical analysis of blockage and optimization of heat transfer performance of fractal-like microchannel nets, *Journal of Electronic Packaging*. Vol. 128, pp. 38–45, 2006.

[156] W. Wechsatol, J.C. Ordonez and S. Kosaraju, Constructal dendritic geometry and the existence of asymmetric bifurcations, *Journal of Applied Physics*. Vol. 100 pp. 113514, 2006.

[157] S. Lorente, Constructal view of electrokinetic transfer through porous media, *Journal of Physics, D Applied. Physic*. Vol. 40, 2941–2947, 2007.

[158] S. Zhou, L. Chen and F. Sun, Optimization of constructal volume-point conduction with variable cross section conducting path, *Energy Conversion and Management*. Vol. 48, pp. 106–111, 2007.

[159] W. Wu, L. Chen and F. Sun, On the "area to point" flow problem based on constructal theory, *Energy Conversion and Management*. Vol. 48, pp. 101–105, 2007.

[160] Y. Azoumah, P. Neveu and N. Mazet, Optimal design of thermochemical reactors based on constructal approach, *AICHE Journal*. Vol. 53, pp. 1257–1266, 2007.

[161] S. Zhou, L. Chen and F. Sun, Constructal entropy generation minimization for heat and mass transfer in a solid–gas reactor based on triangular element, *Journal of Physics. D Applied. Physics*, Vol. 40, pp. 3545–3550, 2007.

[162] W. Wu, L. Chen and F. Sun, Heat-conduction optimization based on Constructal theory, *Applied Energy*, Vol. 84, pp. 39–47, 2007.

[163] V.A.P. Raja, T. Basak and S.K. Das, Thermal performance of a multi-block heat exchanger designed on the basis of Bejan's constructal theory, *International Journal of Heat Mass Transfer*, Vol. 51, pp. 3582–3594, 2008.

[164] P. Xu, B. Yu, S. Qiu and J. Cai, An analysis of the radial flow in the heterogeneous porous media based on fractal and constructal tree networks, *Physica A: Statistical Mechanics and its Applications*, pp. 6471–6483, 2008.

[165] A. Beyene and J. Peffley, Constructal theory, adaptive motion, and their theoretical application to low-speed turbine design, *Journal of Energy Engineering*. Vol. 135, 112–118, 2009.

[166] M.J. Carone, C.B. Williams, J.K. Allen and F. Mistree, An application of constructal theory in the multi-objective design of product platforms, *ASME 2003 Design Engineering Technical Conferences and Computer and Information in Engineering Conference Chicago, Illinois USA, September 2-6*, 2003.

[167] C. Bai and L. Wang, Constructal design of particle volume fraction in nanofluids. *Journal of Heat Transfer*, Vol. 131, pp. 112402-1 - 112402-7, 2009.

[168] F. Wu, L. Chen, X. Kan, A. Shu, K. Wu, Z. Yang. Constructal design of stack filled with parallel plates in standing-wave thermo-acoustic cooler, *Cryogenics* Vol. 49, pp. 107–11, 2009.

[169] C. Bai, and L. Wang, Constructal structure of nanofluids. *Journal of Applied Physics*, Vol. 108, pp. 074317, 2010.

[170] J. Fan, and L. Wang, Constructal design of nanofluids. *International Journal of Heat and Mass Transfer*, Vol. 53, pp. 4238–47, 2010.

[171] S.B. Zhou, L.G Chen and F.R. Sun, Constructal optimization for a solid–gas reactor based on triangular element, *Science China-technological Sciences*, Vol. 51, pp. 1554–62, 2008.

[172] Z. Fan, X. Zhou, L. Luo and W. Yuan. Experimental investigation of the flow distribution of a 2-dimensional constructal distributor, *Experimental Thermal Fluid Science*, Vol. 33, 77–83, 2008.

[173] D. Tondeur, Y. Fan and L Luo, Constructal optimization of arborescent structures with flow singularities, *Chemical Engineering Science*, Vol. 64, pp. 3968–82, 2009.

[174] J. Yue, R. Boichot, L Luo, Y. Gonthier, G. Chen and Q. Yuan, Flow distribution and mass transfer in a parallel microchannel contactor integrated with constructal distributors, *AIChE Journal*, Vol. 56, pp. 298–317, 2010.

[175] J-F Cornet, Calculation of optimal design and ideal productivities of volumetrically lightened photobioreactors using the constructal approach, *Chemical Engineering Science*, Vol. 65, pp. 985–98, 2010

[176] M. Mehrgoo and M. Amidpour, Constructal design of humidification – dehumidification desalination unit architecture, *Desalination*, Vol. 271, pp. 62–71, 2010.

[177] S. Tescari, N. Mazet and P. Neveu, Constructal method to optimize solar thermochemical reactor design, *Solar Energy*, Vol. 84 pp. 1555–66, 2010.

[178] F. Mathieu-Potvin and L. Gosselin, Optimal conduction pathways for cooling a heat generating body: A comparison exercise, *International Journal of Heat Mass Transfer*, Vol. 50, pp. 2996 – 3006, 2007.

[179] Y. Azoumah, P. Neveu and N. Mazet, Constructal design combined with entropy generation minimization for solid-gas reactors, *International Journal of Thermal Science*, Vol. 45, pp. 716–728, 2006.

[180] Y.S. Kim, S. Lorente and A. Bejan, Constructal Steam Generator architecture, *International Journal of Heat Mass Transfer*, Vol. 52, pp. 2362 – 2369, 2009.

[181] Y.S. Kim, S. Lorente and A. Bejan, Distribution of size in steam turbine power plants, *International Journal of Energy Research*, Vol. 33, pp. 989-998, 2011.

[182] S. Bhattacharjee and W.L. Grosshandler, The formation of wall jet near a high temperature wall under microgravity environment, *ASME HTD*, Vol. 96, pp. 711–716, 1998.

[183] S. Petrescu, Comments on the optimal spacing of parallel plates cooled by forced convection, *International Journal of Heat Mass Transfer*, Vol. 37, 1283, 1994.

[184] A. Bejan and S. Lorente, Thermodynamic optimisation of flow geometry in mechanical and civil engineering, J*ournal of non-equilibrum thermodynamic*, Vol. 26, pp. 305 – 354, 2001.

[185] A.H. Ries, Constructal Theory: From Engineering to Physics, and How Flow Systems Develop Shape and Structure, *Applied Mechanics Reviews*, Transactions of the ASME, 2006, Vol. 59, pp. 269–282.

[186] W-P. Ma, S-C. Tzeng and W-J. Jwo, Flow resistance and forced convective heat transfer effects for various flow orientations in a packed channel, *International Communications in Heat and Mass Transfer*, Vol. 33, pp. 319–326, 2006.

[187] X-Q Wang, A. S. Mujumdar and C. Yap, Effect of orientation for phase change material (PCM)-based heat sinks for transient thermal management of electric components, *International Communications in Heat and Mass Transfer*, Vol. 34, pp. 801–808, 2007.

[188] R-T. Huang, W-J. Sheu and C-C. Wang, Orientation effect on natural convective performance of square pin fin heat sinks, *International Journal of Heat and Mass Transfer*, Vol. 51, pp. 2368–2376, 2008.

[189] H. Zhang, I. Mudawar and M. M. Hasan, Experimental and theoretical study of orientation effects on flow boiling CHF, *International Journal of Heat and Mass Transfer*, Vol. 45, pp. 4463–4477, 2002.

[190] P. Dutta, S. Dutta, Effect of baffle size, perforation, and orientation on internal heat transfer enhancement, *International Journal of Heat and Mass Transfer*, Vol. 41, pp. 3005 – 3013, 1998.

[191] A. Tandiroglu, Effect of flow geometry parameters on transient heat transfer for turbulent flow in a circular tube with baffle inserts, *International Journal of Heat and Mass Transfer*, Vol. 49, pp. 1559–1567, 2006.

[192] J.A. Snyman, *Practical mathematical optimisation: an introduction to basic optimisation theory and classical and new gradient-based algorithms*, Springer, New York, 2005.

[193] W.G. Le Roux, Maximum net power output from an intergrated design of a small-scale open and direct solar thermal brayton cycle, M.Eng dissertation, Department of Mechanical and Aeronautical Engineering, University of Pretoria, 2011.

[194] O.A.A. Abdelaziz, Development of multi-scale, Multi-physics, analysis capability and its application to novel heat Exchanger design and Optimization , PhD Thesis, Department of Mechanical Engineering, University of Maryland, College Park, 2009.

[195] A. J. Pacheco-Vega, Simulation of compact heat exchangers using global regression and soft computing, PhD Thesis, Department of Aerospace and Mechanical Engineering, University of Notre Dame, Indiana, 2002.

[196] K. Foli, T. Okabe, M. Olhofer, Y. Jin and B. Sendhoff, Optimization of micro heat exchanger: CFD, analytical approach and multi-objective evolutionary algorithms, *International Journal of Heat and Mass Transfer*, Vol. 49, pp. 1090 – 1099, 2006.

[197] A. Husain and K-Y. Kim, Multiobjective optimisation of a microchannel Heat Sink Using Evolutionary Algorithm, *Journal of Heat Transfer*, Vol. 130, pp. 114505-1-3, 2008.

[198] S. Baodong, W. Lifeng, L. Jianyun and C. Heming, Multi-objective optimization design of a micro-channel heat sink using adaptive genetic algorithm *International Journal of Numerical Methods for Heat and Fluid Flow*, Vol. 21, pp. 353 – 364, 2011.

[199] Fluent Inc., *Fluent Version 6 Manuals*, Centerra Resource Park, 10 Cavendish Court, Lebanon, New Hampshire, USA, 2001 (www.fluent.com).

[200] H.K. Versteeg and W. Malalasekra, *An introduction to computational fluid dynamics: the finite volume method*, 2nd ed., Prentice Hall, England, 2007.

[201] Fluent Inc., *Gambit Version 6 Manuals*, Centerra Resource Park, 10 Cavendish Court, Lebanon, New Hampshire, USA, 2001 (www.fluent.com).

[202] F.M. White, *Viscous Fluid Flow,* 2nd Edition, McGraw-Hill International Editions, Singapore, 1991.

[203] S.V. Patankar, *Numerical Heat Transfer and Fluid flow*, Hemisphere, New York. 1980.

[204] J.A. Snyman, A new dynamic method for unconstrained minimization, *Applied. Mathematical Modelling*, Vol. 6, pp. 449 – 462, 1982.

[205] J.A. Snyman, An improved version of the original leap-frog dynamic method for unconstrained minimization: LFOP1(b), *Applied. Mathematical Modelling*, Vol. 7, pp. 216 – 218, 1983.

[206] J.A. Snyman, N. Stander, and W.J. Roux, A dynamic penalty function method for the solution of structural optimisation problems, *Applied. Mathematical Modelling*, Vol. 18, pp. 453 – 460, 1994.

[207] J.A. Snyman, The LFOPC Leap-frog algorithm for constrained optimisation, *Computer and Mathematics with Applications*, Vol. 40, pp. 1085 – 1096, 2000.

[208] J.A. Snyman, A.M. and Hay, The DYNAMIC-Q optimisation method: an alternative to SQP?, *Computer and Mathematics with Applications*, Vol. 44, pp. 1589–1598, 2002.

[209] J.A. Snyman, and N. Stander, A New successive approximation method for optimum structure design, *AIAA Journal* , Vol. 32, pp. 1310 – 1315, 1994

[210] J.A. Snyman, and N. Stander, Feaseable descent cone methods for inequality constrained optimisation method in engineering, *international journal for numerical engineering*, Vol. 38, pp. 119 – 135, 1995.

[211] D.J. de Kock, Optimal tundish methodology in a continuous casting process, PhD Thesis, Department of Mechanical and Aeronautical Engineering, University of Pretoria, 2005.

[212] D.J. de Kock, Industrial applications of computational flow optimisation. M.Eng dissertation, Department of Mechanical and Aeronautical Engineering, University of Pretoria, 1999.

[213] O.S. Motsamai, Optimisation techniques for combustion designs, PhD Thesis, Department of Mechanical and Aeronautical Engineering, University of Pretoria, 2005.

[214] R.M. Morris , J.A. Snyman and J.P. Meyer, Jets in cross flow mixing analysis using computational fluid dynamics and mathematical optimisation, *Journal Of Propulsion and Power*, Vol. 23, no 14, pp. 1589 – 1598, 2002.

[215] F.U Ighalo, Optimisation of microchannels and micropin-fin heat sinks with computational fluid dynamics in combination with a mathematical optimisation algorithm. M.Eng dissertation, Department of Mechanical and Aeronautical Engineering, University of Pretoria, 2010.

[216] R.C. Chu, Thermal management roadmap cooling electronic products from hand held device to supercomputers Proceedings of MIT Rohsenow Symposium Cambridge, MA, 2002.

[217] SEMATECH The National Technology Roadmap For Semiconductors: Technology Needs, SEMATECH, Austin TX, 1997.

[218] A. Bejan, *Convection Heat Transfer*, third ed., Wiley, Hoboken, 2004.

[219] The MathWorks, Inc., MATLAB & Simulink Release Notes for R2008a, Apple Hill Drive, Natick, MA, 2008 (www.mathworks.com).

A

APPENDIX A: DYNAMIC-Q OPTIMISATION ALGORITHM

A.1 DYNQ.M

```
function [X,F]=dynq(x0,varargin);
tic
%
%          DYNAMIC-Q ALGORITHM FOR CONSTRAINED OPTIMISATION
%               GENERAL MATHEMATICAL PROGRAMMING CODE
%               ------------------------------------
%
% This code is based on the Dynamic-Q method of Snyman documented
% in the paper "THE DYNAMIC-Q OPTIMISATION METHOD: AN ALTERNATIVE
% TO SQP?" by J.A. Snyman and A.M. Hay. Technical Report, Dept Mech.
% Eng., UP.
%
%                 MATLAB implementation by A.M. HAY
%         Multidisciplinary Design Optimisation Group (MDOG)
%  Department of Mechanical Engineering, University of Pretoria
%                       August 2002
%
%                    UPDATED : 23 August 2002
%
%                       BRIEF DESCRIPTION
%                       -----------------
%


%  Dynamic-Q solves inequality and equality constrained optimisation
%  problems of the form:
%
%                    minimise F(X)   ,  X={X(1),X(2),...,X(N)}
%    such that
%                     Cp(X) <= 0        p=1,2,...,NP
%    and
%                     Hq(X) =  0        q=1,2,...,NQ
%    with lower bounds
%        CLi(X) = V_LOWER(i)-X(NLV(i)) <= 0   i=1,2,...,NL
%    and upper bounds
%        CUj(X) = X(NUV(j))-V_UPPER(j) <= 0   j=1,2,...,NU
%
% This is a completely general code - the objective function and the
% constraints may be linear or non-linear. The code therefore solves
% LP, QP and NLP problems.
%
%                       -----------------
%
% User specified functions:
%
% The objective function F and constraint functions C and H must be
```

```
% specified by the user in function FCH. Expressions for the
respective
% gradient vectors must be specified in function GRADFCH.
%
% {The user may compute gradients by finite differences if necessary
%  - see example code in GradFCH}
%
% Side constraints should not be included as inequality constraints
% in the above subroutines, but passed to the dynq function as
% input arguments LO and UP. (Described below)
%
% In addition to FCH and GRADFCH the following functions are called
% by DYNQ and should not be altered:
%       DQLFOPC,DQFUN,DQCONIN,DQCONEQ,DQGRADF,DQGRADC,DQGRADH
%
% In addition the script HISTPLOT.m plots various optimisation
% histories. To suppress automatic plotting set PRNCONST=0 below.
%
%                       -----------------
%
%   synopsis:
%
%           [X,F] = dynq(x0,lo,up,dml,xtol,ftol,clim,np,nq,kloop);
%
%   outputs:
%       X  = optimal solution (1xN)
%       F  = optimal function value
%
%   inputs:
%       x0 = starting point (1xN)
%       lo = NLx2 matrix associated with lower limits on the
variables
%               containing variable index NLV(i) in the first column
and
%               associated value V_LOWER of that limit in the second
column
%               (optional, otherwise assumed no lower side
constraints)
%       up = NUx2 matrix associated with lower limits on the
variables
%               containing variable index NUV(i) in the first column
and
%               associated value V_UPPER of that limit in the second
column
%               (optional, otherwise assumed no upper side
constraints)
%      dml = the move limit which should be approximately the same
order
%               of magnitude as the "radius of the region of
interest"
%               = sqrt(n)*max-variable-range (optional, default =1)
%     xtol = convergence tolerance on the step size (optional,
default =1e-5)
%     ftol = convergence tolerance on the function value (optional,
default =1e-8)
```

```
%     clim = tolerance for determining whether constraints are
violated

%               (optional, default =ftol*1e2)
%       np = number of inequality constraints (optional)
%       nq = number of equality constraints (optional)
%               Note: Both np and nq are optional and determined
automatically
%               if not specified, but at the cost of an extra
function evalution.
%    kloop = maximum number of iterations (optional, default = 100)
%
%     NOTE: use [] to activate default inputs, for example
%
%  [X,F]=dynq(x0,[],[],2); uses dml=2 but default values for all
other inputs.
%
%       See FCH and GRADFCH for an example problem.
%
%   ---- This program is for educational purposes only ----


%*****PLOT OPTIMISATION HISTORIES AT END OF
PROGRAM?******************
%           YES: 1        OR            NO: 0
%
PRNCONST=1;
%*****************************************************************
**

clc;

N=length(x0);   % Determine number of variables
X=x0;

[dum,D]=size(varargin);
vars=cell(1,9);
vars(1:D)=varargin;

LO=vars{1};
UP=vars{2};
DML=vars{3};
XTOL=vars{4};
FTOL=vars{5};
CLIM=vars{6};
NP=vars{7};
NQ=vars{8};
KLOOPMAX=vars{9};

% default values
[NL,dum]=size(LO);
if NL>0
    NLV=LO(:,1)';
    V_LOWER=LO(:,2)';
else
    NLV=[];
```

```
    V_LOWER=[];
end
[NU,dum]=size(UP);

if NU>0
    NUV=UP(:,1)';
    V_UPPER=UP(:,2)';
else
    NUV=[];
    V_UPPER=[];
end
if isempty(DML)
    DML=1; end
if isempty(XTOL)
    XTOL=1e-5; end
if isempty(FTOL)
    FTOL=1e-8; end
if isempty(CLIM)
    CLIM=FTOL*1e2; end
if isempty(NP)|isempty(NQ)
    [F,C,H]=fch(X);
    NP=length(C);
    if isempty(C)
        NP=0;
    end
    NQ=length(H);
    if isempty(H)
        NQ=0;
    end
end
if isempty(KLOOPMAX)
    KLOOPMAX=100; end

%######################################################################
##C
%**********************************************************************
**C
%     MAIN PROGRAM FOLLOWS: Do not alter!!!!
%**********************************************************************
**C
%######################################################################
##C

%*****OPEN OUPUT
FILES************************************************C
%
fidA=fopen('Approx.out','wt+');
fidD=fopen('DynamicQ.out','wt+');
fidH=fopen('History.out','wt+');
%
%*****SPECIFY INITIAL APPROXIMATION
CURVATURES************************C
%
ACURV=0.D0;
BCURV=zeros(1,NP);
if NP==0
    BCURV=[];
```

```
end
CCURV=zeros(1,NQ);
if NQ==0
    CCURV=[];

end
%
%
%
%*****INITIALIZE
OUTPUT***********************************************C
FEASIBLE=0;

fprintf(fidA,' DYNAMICQ OUTPUT FILE \n');
fprintf(fidA,' -------------------- \n');
fprintf(fidA,' Number of variables [N]= %i \n',N);
fprintf(fidA,' Number of inequality constraints [NP]= %i \n',NP);
fprintf(fidA,' Number of equality constraints [NQ]= %i \n',NQ);
fprintf(fidA,' Move limit= %12.8e \n',DML);

fprintf(1,'\n DYNAMICQ OPTIMISATION ALGORITHM \n');
fprintf(1,' ------------------------------ \n');
% (MAXX=Maximum number of X-values to be displayed on screen)
MAXX=4;
if N<=MAXX
    fprintf(1,' Iter Function value  ? XNORM      RFD        ');
    fprintf(1,'X(%i)       ',1:N);
    fprintf(1,'\n ----------------------------------------');
    for I=1:N
        fprintf(1,'------------',1:N);
    end
    fprintf(1,'\n');
else
    fprintf(1,' Iter Function value  ? XNORM      RFD ');
    fprintf(1,'\n ----------------------------------------\n');
end

fprintf(fidD,' DYNAMICQ OPTIMISATION ALGORITHM\n');
fprintf(fidD,' ------------------------------\n');
fprintf(fidD,' Iter Function value       ? XNORM      RFD        ');
fprintf(fidD,'X(%i)           ',1:N);
fprintf(fidD,'\n');

fprintf(fidD,' ------------------------------------------------');
for i=1:N
    fprintf(fidD,'---------------');
end
fprintf(fidD,'\n');

% Initialize outer loop counter
KLOOP=0;

% Arbitrary large values to prevent premature termination
F_LOW=1.D6;
RFD=1.D6;
RELXNORM=1.D6;
```

```
C_A=zeros(1,NP+NL+NU+1);

%*****START OF OUTER OPTIMISATION
LOOP*******************************C

while KLOOP<=KLOOPMAX

%*****APPROXIMATE
FUNCTIONS*****************************************C

% Determine function values
    [F,C,H]=fch(X);

% Calculate relative step size
    if KLOOP>0
        DELXNORM=sqrt((X_H(KLOOP,:)-X)*(X_H(KLOOP,:)-X)');
        XNORM=sqrt(X*X');
        RELXNORM=DELXNORM/(1+XNORM);
    end

% Determine lowest feasible function value so far
    if KLOOP>0
    FEASIBLE=1;
        check=find(C<CLIM);
        if isempty(check)&NP>0;
            FEASIBLE=0;
        end
        check=find(abs(H)<CLIM);
        if isempty(check)&NQ>0;
            FEASIBLE=0;
        end
        for I=1:NL
            if C_A(I+NP)>CLIM
                FEASIBLE=0;
            end
        end
        for I=1:NU
            if C_A(I+NP+NL)>CLIM
                FEASIBLE=0;
            end
        end
    end

% Calculate relative function difference
    if F_LOW~=1.D6&FEASIBLE==1
        RFD=abs(F-F_LOW)/(1+abs(F));
    end

    if FEASIBLE==1&F<F_LOW
        F_LOW=F;
    end

% Store function values
    X_H(KLOOP+1,:)=X;   % Need to adjust from Fortran version since
```

```
    F_H(KLOOP+1)=F;     % Matlab does not accept 0 as a matrix index
    if NP>0
        C_H(KLOOP+1,1:NP)=C;
    end
    if NL>0
        C_H(KLOOP+1,NP+1:NP+NL)=C_A(NP+1:NP+NL);

    end
    if NU>0
        C_H(KLOOP+1,NP+NL+1:NP+NL+NU)=C_A(NP+NL+1:NP+NL+NU);
    end
    C_H(KLOOP+1,NP+NL+NU+1)=C_A(NP+NL+NU+1);
    if NQ>0
        H_H(KLOOP+1,:)=H;
    end

% Determine gradients
    [GF,GC,GH]=gradfch(X);

% Calculate curvatures
    if KLOOP>0
        DELX=X_H(KLOOP,:)-X_H(KLOOP+1,:);
        DELXNORM=DELX*DELX';

% Calculate curvature ACURV
        DP=GF*DELX';
        ACURV=2.*(F_H(KLOOP)-F_H(KLOOP+1)-GF*DELX')/DELXNORM;

        for J=1:NP
            DP=GC(J,:)*DELX';
% Calculate corresponding curvature BCURV(J)
            BCURV(J)=2.*(C_H(KLOOP,J)-C_H(KLOOP+1,J)-
GC(J,:)*DELX')/DELXNORM;
        end

        for J=1:NQ
            DP=GH(J,:)*DELX';
% Calculate corresponding curvature CCURV(J)
            CCURV(J)=2.*(H_H(KLOOP,J)-H_H(KLOOP+1,J)-
GH(J,:)*DELX')/DELXNORM;
        end
    end

%*****RECORD PARAMETERS FOR THE
ITERATION*****************************C

% Write approximation constants to Approx.out
    fprintf(fidA,' Iteration %i \n',KLOOP);
    fprintf(fidA,' --------------\n');
    fprintf(fidA,' X=\n');
    for I=1:N
        fprintf(fidA,' %12.8f ',X(I));
    end
    fprintf(fidA,'\n F= %15.8e\n',F);
    for I=1:NP
        fprintf(fidA,' C(%i)=%15.8e',I,C(I));
```

```
    end
    for I=1:NQ
         fprintf(fidA,' H(%i)=%15.8e',I,H(I));
    end

    fprintf(fidA,' Acurv=%15.8e',ACURV);
    for I=1:NP

     fprintf(fidA,' Bcurv(%i)=%15.8e',I,BCURV(I));
    end
    for I=1:NQ
        fprintf(fidA,' Ccurv(%i)=%15.8e',I,CCURV(I));
    end

% Write solution to file
    if KLOOP==0
        fprintf(fidD,' %4i %+19.12e %i
',KLOOP,F,FEASIBLE);
    else
        if RFD~=1.D6
            fprintf(fidD,' %4i %+19.12e %i %9.3e
%9.3e',KLOOP,F,FEASIBLE,RELXNORM,RFD);
        else
            fprintf(fidD,' %4i %+19.12e %i %9.3e
',KLOOP,F,FEASIBLE,RELXNORM);
        end
    end
    fprintf(fidD,' %+13.6e',X);
    fprintf(fidD,'\n');

% Write solution to screen
    if KLOOP==0
        if N<=MAXX
            fprintf(1,' %4i %+14.7e %i
',KLOOP,F,FEASIBLE);
            fprintf(1,' %+9.2e',X);
            fprintf(1,'\n');
        else
            fprintf(1,' %4i %+14.7e %i\n',KLOOP,F,FEASIBLE);
        end
     else
        if N<=MAXX
            if RFD~=1.D6&FEASIBLE==1
                fprintf(1,' %4i %+14.7e %i %9.3e
%9.3e',KLOOP,F,FEASIBLE,RELXNORM,RFD);
            else
                fprintf(1,' %4i %+14.7e %i %9.3e
',KLOOP,F,FEASIBLE,RELXNORM);
            end
            fprintf(1,' %+9.2e',X);
            fprintf(1,'\n');
        else
            if RFD~=1.D6&FEASIBLE==1
                fprintf(1,' %4i %+14.7e %i %9.3e
%9.3e\n',KLOOP,F,FEASIBLE,RELXNORM,RFD);
            else
```

```
                fprintf(1,' %4i %+14.7e %i
%9.3e\n',KLOOP,F,FEASIBLE,RELXNORM);
            end
        end
    end

% Exit do loop here on final iteration
    if KLOOP==KLOOPMAX|RFD<FTOL|RELXNORM<XTOL
        if KLOOP==KLOOPMAX

        fprintf(1,' Terminated on max number of steps\n');
            fprintf(fidD,' Terminated on max number of steps\n');
        end
        if RFD<FTOL
            fprintf(1,' Terminated on function value\n');
            fprintf(fidD,' Terminated on function value\n');
        end
        if RELXNORM<XTOL
            fprintf(1,' Terminated on step size\n');
            fprintf(fidD,' Terminated on step size\n');
        end
        fprintf(1,'\n');
        fprintf(fidD,'\n');
        break;
    end

%*****SOLVE THE APPROXIMATED
SUBPROBLEM*******************************C

[X,F_A,C_A,H_A]=dqlfopc(X,NP,NQ,F,C,H,GF,GC,GH,ACURV,BCURV,CCURV,DML.
..
    ,NL,NU,NLV,NUV,V_LOWER,V_UPPER,XTOL,KLOOP);

% Record solution to approximated problem

    fprintf(fidA,'Solution of approximated problem:\n');
    fprintf(fidA,'X=\n');
    for I=1:N
        fprintf(fidA,' %12.8f\n',X(I));
    end
    fprintf(fidA,' F_A=%15.8e\n',F_A);
    for I=1:NP+NL+NU+1
        fprintf(fidA,'C_A(%i)=%15.8e\n',I,C_A(I));
    end
    for I=1:NQ
        fprintf(fidA,'H_A(%i)=%15.8e\n',I,H_A(I));
    end

% Increment outer loop counter
    KLOOP=KLOOP+1;
end

% Write final constraint values to file

if NP>0
    fprintf(fidD,' Final inequality constraint function values:\n');
```

```
    for I=1:NP
        fprintf(fidD,'  C(%i)=%15.8e\n',I,C(I));
    end
end
if NQ>0
    fprintf(fidD,' Final equality constraint function values:\n');
    for I=1:NQ
        fprintf(fidD,'  H(%i)=%15.8e\n',I,H(I));
    end
end

if NL>0
    fprintf(fidD,' Final side (lower) constraint function
values:\n');
    for I=1:NL
        fprintf(fidD,'  C(X(%i))=%15.8e\n',NLV(I),C_A(NP+I));
    end
end
if NU>0
    fprintf(fidD,' Final side (upper) constraint function
values:\n');
    for I=1:NU
        fprintf(fidD,'  C(X(%i))=%15.8e\n',NUV(I),C_A(NP+NL+I));
    end
end

% Write final constraint values to screen
fprintf(1,' Constraint values follow:\n\n')
if NP>0
    fprintf(1,' Final inequality constraint function values:\n');
    for I=1:NP
        fprintf(1,'  C(%i)=%15.8e\n',I,C(I));
    end
end
if NQ>0
    fprintf(1,' Final equality constraint function values:\n');
    for I=1:NQ
        fprintf(1,'  H(%i)=%15.8e\n',I,H(I));
    end
end
if NL>0
    fprintf(1,' Final side (lower) constraint function values:\n');
    for I=1:NL
        fprintf(1,'  C(X(%i))=%15.8e\n',NLV(I),C_A(NP+I));
    end
end
if NU>0
    fprintf(1,' Final side (upper) constraint function values:\n');
    for I=1:NU
        fprintf(1,'  C(X(%i))=%15.8e\n',NUV(I),C_A(NP+NL+I));
    end
end

% Write history vectors

fprintf(fidH,' %3i%3i%3i%3i%3i%3i\n', KLOOP,N,NP,NL,NU,NQ);
for I=1:KLOOP+1
```

```
    fprintf(fidH,' %3i %15.8e',I-1,F_H(I));
    for J=1:N
        fprintf(fidH,' %15.8e',X_H(I,J));
    end
    fprintf(fidH,'\n');
end
if NP>0
    for I=1:KLOOP+1
        fprintf(fidH,' %3i',I-1);
        for J=1:NP
            fprintf(fidH,' %15.8e',C_H(I,J));

      end
        fprintf(fidH,'\n');
    end
end
if NL>0
    for I=1:KLOOP+1
        fprintf(fidH,' %3i',I-1);
        for J=NP+1:NP+NL
            fprintf(fidH,' %15.8e',C_H(I,J));
        end
        fprintf(fidH,'\n');
    end
end
if NU>0
    for I=1:KLOOP+1
        fprintf(fidH,' %3i',I-1);
        for J=NP+NL+1:NP+NL+NU
            fprintf(fidH,' %15.8e',C_H(I,J));
        end
        fprintf(fidH,'\n');
    end
end
if NQ>0
    for I=1:KLOOP+1
        fprintf(fidH,' %3i',I-1);
        for J=1:NQ
            fprintf(fidH,' %15.8e',H_H(I,J));
        end
        fprintf(fidH,'\n');
    end
end

fclose(fidD);
fclose(fidH);
fclose(fidA);

if PRNCONST
    histplot;
%    disp('Press a key to continue');
%    pause;
%    close all;
end
toc
```

A.2 FCH.M

```
function [F,C,H]=fch(X);
% Objective and constraint function evaluation for DYNAMIC-Q
%           (USER SPECIFIED)
%
%   synopsis:
%
%       [F,C,H]=fch(X);
%
%   outputs:
%       F  = objective function value
%       C  = vector of inequality constraint functions (1xNP)
%       H  = vector of equality constraint functions (1xNQ)
%
%   inputs:
%       X = design vector (1xN)
%
%                       -----------------
%
% The application of the code is illustrated here for the very simple
% but general example problem (Hock 71):
%
%      minimise  F(X) = X(1)*X(4)*(X(1)+X(2)+X(3))+X(3)
% such that
%                 C(X) = 25-X(1)*X(2)*X(3)*X(4) <= 0
%      and
%                 H(X) = X(1)^2+X(2)^2+X(3)^2+X(4)^2-40 = 0
%
%     and side constraints
%
%                 1 <= X(I) <= 5 , I=1,2,3,4
%
%   Starting point is (1,5,5,1)
%
%   Solution of this problem is accomplished by:
%       (with FCH and GRADFCH unaltered)
%
%       x0=[1,5,5,1] % Specify starting point
%       lo=[1:4;1,1,1,1]' % Specify lower limits
%       up=[1:4;5,5,5,5]' % Specify upper limits
%       [X,F]=dynq(x0,lo,up); % Solve using Dynamic-Q
%
%   NOTE: This function should return C=[]; H=[]; if these are
%         not defined.
%
%   See also DYNQ and GRADFCH
%

%%My programme in FCH

%-----------------------------------------------------------------------

% x1 is the hydraulic diameter  of the elemental channel
```

```
% x2 is the height or width  of the elemental volume

%x = [ 0.00005*0.8^0.5    0.00005]


fid = fopen('designvariable.jou','w');

for i=1:2

fprintf(fid,'$x%g = %g\n',i,X(i));

end

  fclose(fid);

%-----------------------------------------------------------------------
-----

!cd C:\FLUENT PRACTICE 3\CYLINDER\CYLINDER DYNQ

!designlink.bat

!Gabitlink.bat

!Fluentlink.bat

fid = fopen('CylinderTemp.dat', 'r');

CylinderTemp = textscan(fid, '%f %f %f %f %f ', 20000, 'headerlines',
1);


Tmax = max(CylinderTemp{:,5})-273.15;


fclose(fid);

F = Tmax';

%w = X(1) + 2*X(2) ;

POROSITY = (X(1)/X(2))^2.;

%POROSITY = (X(1)/(X(1)+2*X(2)))^2;

   X  ,      POROSITY,     F
%-------------------------------------------------------

%Inequality Constraints
```

```
%using dh and h FOR C
%------------------------------------------------------

 % porosity
%------------------------------------------------------

%C(1)= 0.2 - (X(1)/X(2))^2;


%C(2)= (X(1)./X(2)).^2 - 0.8;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

%C(1)= 0.05.*X(2)^2 - X(1)^2;

%C(2)= X(1)^2 - 0.1.*X(2)^2;

C(1)= 0.05.*X(2)^2 - X(1)^2;

C(2)= X(1)^2 - 0.2.*X(2)^2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

%C = [];

%H(1)=X(1)^2 - 0.2.*X(2)^2;

%H(1)=X(1)^2 - 0.1.*X(2)^2;

% h > d  value
%------------------------------------------------------

%C(3)= X(1) - X(2);

%------------------------------------------------------

%Equality Constraints
%H(1)=(X(1)*X(2)/0.01)-1;

% To eliminate error messages
% Do not delete

if ~exist('C')
    C=[];
end
if ~exist('H')
    H=[];
end
```

```
%-------------------------------------------------------------------
-----
```

A.3 GRADFCH.M

```
function [GF,GC,GH]=gradfch(X);
% Objective and constraint function GRADIENT evaluation for DYNAMIC-Q
%           (USER SPECIFIED)
%
%   synopsis:
%
%       [GF,GC,GH]=gradfch(X);
%
%   outputs: Partial derivatives wrt variables X(I) of
%      GF  = objective function (1xN)
%      GC  = inequality constraint functions (NPxN)
%      GH  = equality constraint functions (NQxN)
%
%   inputs:
%       X = design vector (1xN)
%
%     COMPUTE THE GRADIENT VECTORS OF THE OBJECTIVE FUNCTION F,
%     INEQUALITY CONSTRAINTS C, AND EQUALITY CONSTRAINTS H
%     W.R.T. THE VARIABLES X(I):
%              GF(I),I=1,N
%              GC(J,I), J=1,NP I=1,N
%              GH(J,I), J=1,NQ I=1,N
%
%   NOTE: This function should return GC=[]; GH=[]; if these are
%         not defined.
%
%   See also DYNQ, FCH
%

% Determine gradients by finite difference
FDFLAG=1;

if FDFLAG
    DELTX=1.D-4;    % Finite difference interval
    [F,C,H]=fch(X);
    N=length(X);
    for I=1:N
        DX=X;
        DX(I)=X(I)+DELTX;
        [F_D,C_D,H_D]=fch(DX);
        GF(I)=(F_D-F)/DELTX;
        if ~isempty(C)
              GC(1,1)= - 2.*X(1);


     GC(1,1)= - 2.*X(1);

     GC(1,2)=  0.1.*X(2);
```

```
 GC(2,1)=  2.*X(1);

     GC(2,2)= -0.4.*X(2);


   end

        end
    end
end

% To eliminate error messages
% Do not erase
if ~exist('GC')
    GC=[];
end
if ~exist('GH')
    GH=[];
end
```

A.4 RUNDYNQ

```
%This program initiates DYNQ.M

clear all

clc

tic

dml = 0.005;
xtol = 1e-8;
ftol = 1e-6;
clim = ftol;
np = 2;
nq = [];
kloop = 100;

%-----------------------------------------------------------------------
------

x0 = [0.25     0.4    ]; % Specify starting point 40kbc 1.238774e-001

lo = [1:2; 0.08 , 0.2  ]'; %Specify lower limits   %(Note the use of
the transpose)

up = [1:2; 0.25 , 0.8  ]'; %Specify upper limits %(Note the use of
the transpose)

%  Synopsis: % Solve using Dynamic-Q

[X,F] = dynq(x0,lo,up,dml,xtol,ftol,clim,np,nq,kloop);

toc
```

B

APPENDIX B: GAMBIT JOURNAL FILE FOR GEOMETRY AND MESH GENERATION OF COOLING CHANNELS

B.1 CIRCULAR COOLING CHANNELS JOURNAL FILE

```
/-----------------------------------------------------------------/

/Parameter
/////

/-----------------------------------------------------------------/
/read the dimension of the brick volume 1

/$v = ((( $x )^2* (0.01))^(1/3))

/$w1 = ( $x1 + 2*$x2)/1000

$w1 = $x2/1000


/$w1 = $x/$v

/$w1 = $x

/$w1 = $x/1000

$d1 = $w1

$h1 = 0.01

/-----------------------------------------------------------------/

/read the dimension of the cylinder volume 2

$r1=$x1/2/1000

/$r1=$dh/2/$v

/$r1=$dh

/$r1=$dh/1000

$r3=$r1
```

```
$h1 = 0.01

/-----------------------------------------------------------------/

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1
$offsety1 = 0.5*$d1
$offsetz1 =-0.5*$h1

/-----------------------------------------------------------------/
/read the offset origin of cylinder volume 2

$offsetcylh1 = 0
$offsetcylr1 = 0
$offsetcylr3 =-0.5*$h1

/-----------------------------------------------------------------/

/move by offset cylinder volume 2

$movex1 = 0.5*$w1
$movey1 = 0.5*$w1

$movez1 = 0

/-----------------------------------------------------------------/
/mesh brick V1

/$meshv1=0.12*$w1

$meshv1=0.1*$w1

/$meshv1=0.08*$w1

/$meshv1=$mesh1

/-----------------------------------------------------------------/
/mesh cylinder V2

$meshv2=0.1*$x1/1000

/$meshv2=0.1*2*$r1


/$meshv2=0.07*2*$r1

$meshv2=0.1*$dh

/$meshv2=$mesh2

/-----------------------------------------------------------------/

volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 $offsetz1 brick
```

```
volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1
$offsetcylr1 $offsetcylr3 zaxis frustum
volume move "volume.2" offset $movex1 $movey1 $movez1
/-----------------------------------------------------------------/
volume subtract "volume.1" volumes "volume.2" keeptool
face connect "face.7" "face.8" "face.9" "face.10" real
/-----------------------------------------------------------------/
volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1
volume mesh "volume.2" cooper source "face.7" "face.9" size $meshv2
/-----------------------------------------------------------------/
physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"
physics create "symmetry top" btype "SYMMETRY" face "face.5"
physics create "symmetry bottom" btype "SYMMETRY" face "face.2"
physics create "symmetry left" btype "SYMMETRY" face "face.3"
physics create "symmetry right" btype "SYMMETRY" face "face.4"
physics create "pressure inlet" btype "PRESSURE_INLET" face "face.9"
physics create "pressure outlet" btype "PRESSURE_OUTLET" face
"face.7"
physics create "wall internal" btype "WALL" face "face.8"
physics create "solid" ctype "SOLID" volume "volume.1"
physics create "fluid" ctype "FLUID" volume "volume.2"

/-----------------------------------------------------------------/

save name "C:\\FLUENT PRACTICE 3\\CYLINDER\\CYLINDER
DYNQ\\CylinderDYNQDW.dbs"


export fluent5 \
  "C:\\FLUENT PRACTICE 3\\CYLINDER\\CYLINDER
DYNQ\\CylinderDYNQDW.msh"

save

/-----------------------------------------------------------------/
```

B.2 SQUARE COOLING CHANNELS JOURNAL FILE

```
/-----------------------------------------------------------------/

/Parameter

/-----------------------------------------------------------------/

/read the dimension of the brick volume 1

$w1 = $x2/1000

/$w1 = $x

$d1 = $w1

$h1 = 0.01

/-----------------------------------------------------------------/

/read the dimension of the brick volume 2

$w2=$x1/1000

/$w2=$dh

$d2=$w2

$h1 = 0.01

/-----------------------------------------------------------------/

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1

$offsety1 = 0.5*$d1

$offsetz1 =-0.5*$h1

/-----------------------------------------------------------------/

/read the offset origin of brick volume 2

$offsetx2 = 0.5*$w2

$offsety2 = 0.5*$d2

$offsetz2 =-0.5*$h1


/-----------------------------------------------------/

/move by offset brick volume 2

$movex2 = 0.5*($w1-$w2)
```

```
$movey2 = 0.5*($d1-$d2)

$movez2 = 0

/-----------------------------------------------------------------/

/mesh brick V1


$meshv1=0.08*$w1

/$meshv1=0.2*$w1

/$meshv1=$mesh1

/-----------------------------------------------------------------/
/mesh brick V2


/$meshv2=0.07*$w2

$meshv2=0.2*$w2

/$meshv2=$mesh2

/-----------------------------------------------------------------/

volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 $offsetz1 brick


volume create width $w2 depth $d2 height $h1 offset $offsetx2

$offsety2 $offsetz2 brick

/-----------------------------------------------------------------/

volume move "volume.2" offset $movex2 $movey2 $movez2

volume subtract "volume.1" volumes "volume.2" keeptool

/---------------------------------------------------------/

face connect "face.7" "face.8" "face.9" "face.10" "face.11" "face.12"

\ "face.15" "face.16" "face.17" "face.18" real

/-----------------------------------------------------------------/

volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.7" "face.12" size $meshv2
```

```
/-------------------------------------------------------------------/

physics create "wall front" btype "WALL" face "face.6"

physics create "wall back" btype "WALL" face "face.1"

physics create "symmetry top" btype "SYMMETRY" face "face.5"

physics create "symmetry bottom" btype "SYMMETRY" face "face.2"

physics create "symmetry left" btype "SYMMETRY" face "face.3"

physics create "symmetry right" btype "SYMMETRY" face "face.4"

physics create "wall top internal" btype "WALL" face "face.11"

physics create "wall bottom internal" btype "WALL" face "face.8"

physics create "wall left internal" btype "WALL" face "face.9"

physics create "wall right internal" btype "WALL" face "face.10"

physics create "Pressure inlet" btype "PRESSURE_INLET" face "face.12"

physics create "Pressure outlet" btype "PRESSURE_OUTLET" face

"face.7"

physics create "fluid" ctype "FLUID" volume "volume.2"

physics create "solid" ctype "SOLID" volume "volume.1"

/-------------------------------------------------------------------/

save name "C:\\FLUENT PRACTICE 3\\SQUARE\\SQUARE
DYNQ\\SquareDYNQDW.dbs"

export fluent5 "C:\\FLUENT PRACTICE 3\\SQUARE\\SQUARE
DYNQ\\SquareDYNQDW.msh"

save

/-------------------------------------------------------------------/
```

B.3 ISOCELES RIGHT TRIANGULAR COOLING CHANNELS

JOURNAL FILE

```
/-------------------------------------------------------------------/

/Parameter
/////

/read the dimension of the Brick volume 1
```

```
/$w1 = ( $x1 + $x2)


$w1 = $x2/1000

/$w1 = $w

$d1 = $w1

$h1 = 0.01

/-----------------------------------------------------------------/

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1
$offsety1 = 0.5*$d1
$offsetz1 = 0.5*$h1

/-----------------------------------------------------------------/

/read the dimension of the Triangle volume 2

/-----------------------------------------------------------------/

/$hT = $hT

/$dh = $dh

$a = $x1/1000

$hT = $a/2

$dh = $a^2/($a + (2)^0.5*$a )

/-----------------------------------------------------------/
vertice or point 1


$vertx1 =  0
$verty1 =  0

$vertz1  = 0

/-----------------------------------------------------------/

vertice or point 2

$vertx2 =  $a
$verty2 =  0
$vertz2  = 0

/-----------------------------------------------------------/
vertice or point 3

$vertx3 =  0.5*$a
```

```
$verty3 =  $hT
$vertz3  = 0

/-----------------------------------------------------------------/

/move by  offset Triangle volume 2

$movex1 = 0.5*($w1-$a)

$movey1 = 0.5*($w1-$hT)

$movez1 = 0

/-----------------------------------------------------------------/

/mesh brick V1

$meshv1=0.09*$w1

/-------------------------------------------------

/mesh Triangle  V2

/$meshv2=0.1*$x1

/$meshv2=0.1*$w1

$meshv2=0.09*$dh

/-----------------------------------------------------------------/
volume create width $w1 depth $d1 height $h1 offset $offsetx1
$offsety1 \
  -$offsetz1 brick

/-----------------------------------------------------------------/

vertex create coordinates $vertx1 $verty1 $vertz1

vertex create coordinates $vertx2 $verty2 $vertz2


vertex create coordinates $vertx3 $verty3 $vertz3

/-----------------------------------------------------------/

edge create straight "vertex.9" "vertex.10"
edge create straight "vertex.10" "vertex.11"
edge create straight "vertex.9" "vertex.11"

/-----------------------------------------------------------------/
face create wireframe "edge.13" "edge.14" "edge.15" real

/-----------------------------------------------------------------/
volume create translate "face.7" vector 0 0 -$h1

/-----------------------------------------------------------------/
```

```
volume move "volume.2" offset $movex1 $movey1 $movez1

/-----------------------------------------------------------------/

volume subtract "volume.1" volumes "volume.2" keeptool

/-----------------------------------------------------------------/

face connect "face.1" "face.2" "face.3" "face.4" "face.5" "face.6"
"face.7" \
  "face.8" "face.9" "face.10" "face.11" "face.12" "face.13" "face.14"
real

/-----------------------------------------------------------------/

face link "face.2" "face.5" edges "edge.5" "edge.12" vertices
"vertex.5" \
  "vertex.7" reverse periodic

/-----------------------------------------------------------------/

volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.8" "face.7" size $meshv2

/-----------------------------------------------------------------/

physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"
physics create "wall internal left" btype "WALL" face "face.10"

physics create "wall internal right" btype "WALL" face "face.11"

physics create "wall internal bottom" btype "WALL" face "face.9"

physics create "symmetry left" btype "SYMMETRY" face "face.3"

physics create "symmetry right" btype "SYMMETRY" face "face.4"

physics create "Periodic BT" btype "PERIODIC" face "face.2" "face.5"

physics create "pressure inlet" btype "PRESSURE_INLET" face "face.7"

physics create "pressure outlet" btype "PRESSURE_OUTLET" face

"face.8"

physics create "solid" ctype "SOLID" volume "volume.1"

physics create "fluid" ctype "FLUID" volume "volume.2"


/-----------------------------------------------------------------/

save name \
```

```
  "C:\\FLUENT PRACTICE 3\\TRIANGLE\\TRIANGLE
DYNQPeriodic45\\TriangleDYNQPeriodic45DW.dbs"

export fluent5 \
  "C:\\FLUENT PRACTICE 3\\TRIANGLE\\TRIANGLE
DYNQPeriodic45\\TriangleDYNQPeriodic45DW.msh"

save

/------------------------------------------------------------------/
```

B.4 EQUILATERAL TRIANGULAR COOLING CHANNELS

JOURNAL FILE

```
/------------------------------------------------------------------/

/Parameter

/read the dimension of the Brick volume 1

/$w1 = ( $x1 + $x2)

 $w1 = $x2/1000

/$w1 = $w

 $d1 = $w1

 $h1 = 0.01

/------------------------------------------------------------------/

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1
$offsety1 = 0.5*$d1
$offsetz1 = 0.5*$h1

/------------------------------------------------------------------/

/read the dimension of the Triangle volume 2


/$a = $x1/1000


/$a = $x1/1000

$hT = (3)^0.5/2*$a

$dh = $a/(3)^0.5
```

```
/vertice or point 1


$vertx1 =  0
$verty1 =  0
$vertz1  = 0

/-------------------------------------------------------------/

/vertice or point 2

$vertx2 =  $a
$verty2 =  0
$vertz2  = 0


/-------------------------------------------------------------------/
/vertice or point 3

$vertx3 =  0.5*$a
$verty3 =  $hT
$vertz3  = 0

/-------------------------------------------------------------------/

/move by offset Triangle volume 2

$movex1 = 0.5*($w1-$a)

$movey1 = 0.5*($w1-$hT)

$movez1 = 0

/-------------------------------------------------------------------/

/mesh brick V1

$meshv1=0.09*$w1

/mesh Triangle V2

/$meshv2=0.1*$x1

/$meshv2=0.1*$w1

$meshv2=0.09*$dh

/-------------------------------------------------------------------/

volume create width $w1 depth $d1 height $h1 offset $offsetx1
$offsety1 \
  -$offsetz1 brick

/-------------------------------------------------------------------/

vertex create coordinates $vertx1 $verty1 $vertz1
```

```
vertex create coordinates $vertx2 $verty2 $vertz2

vertex create coordinates $vertx3 $verty3 $vertz3

/-----------------------------------------------------------------/

edge create straight "vertex.9" "vertex.10"
edge create straight "vertex.10" "vertex.11"
edge create straight "vertex.9" "vertex.11"

/-----------------------------------------------------------------/

face create wireframe "edge.13" "edge.14" "edge.15" real

/-----------------------------------------------------------------/

volume create translate "face.7" vector 0 0 -$h1
/-----------------------------------------------------------------/
volume move "volume.2" offset $movex1 $movey1 $movez1

/-----------------------------------------------------------------/

volume subtract "volume.1" volumes "volume.2" keeptool

/-----------------------------------------------------------------/

face connect "face.1" "face.2" "face.3" "face.4" "face.5" "face.6"
"face.7" \
  "face.8" "face.9" "face.10" "face.11" "face.12" "face.13" "face.14"
real

/-----------------------------------------------------------------/

face link "face.2" "face.5" edges "edge.5" "edge.12" vertices
"vertex.5" \
  "vertex.7" reverse periodic

/-----------------------------------------------------------------/

volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.8" "face.7" size $meshv2

/-----------------------------------------------------------------/

physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"

physics create "wall internal left" btype "WALL" face "face.10"

physics create "wall internal right" btype "WALL" face "face.11"


physics create "wall internal bottom" btype "WALL" face "face.9"

physics create "symmetry left" btype "SYMMETRY" face "face.3"
```

```
physics create "symmetry right" btype "SYMMETRY" face "face.4"

physics create "Periodic BT" btype "PERIODIC" face "face.2" "face.5"

physics create "pressure inlet" btype "PRESSURE_INLET" face "face.7"

physics create "pressure outlet" btype "PRESSURE_OUTLET" face

"face.8"

physics create "solid" ctype "SOLID" volume "volume.1"

physics create "fluid" ctype "FLUID" volume "volume.2"


/-----------------------------------------------------------------/

save name \
  "C:\\FLUENT PRACTICE 3\\TRIANGLE\\TRIANGLE
DYNQPeriodic60\\TriangleDYNQPeriodicDW.dbs"


export fluent5 \
  "C:\\FLUENT PRACTICE 3\\TRIANGLE\\TRIANGLE
DYNQPeriodic60\\TriangleDYNQPeriodicDW.msh"

save

/-----------------------------------------------------------------/
```

B.5 RECTANGULAR COOLING CHANNELS JOURNAL FILE

```
/-----------------------------------------------------------------/

/Parameter
/////


/read the dimension of the brick volume 1


/$w1 = $Wex

/$d1 = $Hex

$w1 = $x1/1000

$d1 = $x2/1000

/$w1 = $x1

/$d1 = $x2
```

```
/$DHex = (2*($x1*$x2))/($x1 + $x2)/1000

$h1 = 0.01

/-----------------------------------------------------------------/

/read the dimension of the brick volume 2

/$w2 = $win

/$d2 = $hin

$w2 = $x3/1000

$d2 = $x4/1000

/$w2 = $x3

/$d2 = $x4

/$dhin = (2*($x3*$x4))/($x3 + $x4)/1000

$h1 = 0.01

/-----------------------------------------------------------------/

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1

$offsety1 = 0.5*$d1

$offsetz1 =-0.5*$h1

/read the offset origin of brick volume 2

$offsetx2 = 0.5*$w2

$offsety2 = 0.5*$d2

$offsetz2 =-0.5*$h1

/-----------------------------------------------------------/

/move by offset brick volume 2

$movex2 = 0.5*($w1-$w2)

$movey2 = 0.5*($d1-$d2)

$movez2 = 0


/-----------------------------------------------------------------/

/mesh brick V1
```

```
$meshv1=0.09*$DHex

/$meshv1=0.2*($DHex - $dhin)

/$meshv1=0.12*(2*($x1*$x2))/($x1 + $x2)/1000

/$meshv1=0.12*(2*($x1*$x2))/($x1 + $x2)

/$meshv1=0.12*$Hex

/$meshv1=0.12*$Wex

/-----------------------------------------------------------------/

/mesh brick V2

$meshv2=0.09*$dhin

/$meshv2=0.1*(2*($x3*$x4))/($x3 + $x4)/1000

/$meshv2=0.1*(2*($x3*$x4))/($x3 + $x4)

/$meshv2=0.1*$win

/-----------------------------------------------------------------/

volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 $offsetz1 brick

volume create width $w2 depth $d2 height $h1 offset $offsetx2

$offsety2 $offsetz2 brick

/-----------------------------------------------------------------/

volume move "volume.2" offset $movex2 $movey2 $movez2

volume subtract "volume.1" volumes "volume.2" keeptool

/-----------------------------------------------------------------/

face connect "face.7" "face.8" "face.9" "face.10" "face.11" "face.12"
\
  "face.15" "face.16" "face.17" "face.18" real

/-----------------------------------------------------------------/

volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.7" "face.12" size $meshv2

/-----------------------------------------------------------------/

physics create "wall front" btype "WALL" face "face.6"
```

```
physics create "wall back" btype "WALL" face "face.1"

physics create "symmetry top" btype "SYMMETRY" face "face.5"

physics create "symmetry bottom" btype "SYMMETRY" face "face.2"

physics create "symmetry left" btype "SYMMETRY" face "face.3"

physics create "symmetry right" btype "SYMMETRY" face "face.4"

physics create "wall top internal" btype "WALL" face "face.11"

physics create "wall bottom internal" btype "WALL" face "face.8"

physics create "wall left internal" btype "WALL" face "face.9"

physics create "wall right internal" btype "WALL" face "face.10"

physics create "pressure inlet" btype "PRESSURE_INLET" face "face.12"

physics create "pressure outlet" btype "PRESSURE_OUTLET" face

"face.7"

/-----------------------------------------------------------/

physics create "solid" ctype "SOLID" volume "volume.1"

physics create "fluid" ctype "FLUID" volume "volume.2"


/-----------------------------------------------------------/

save name \
  "C:\\FLUENT PRACTICE 3\\RECTANGLE\\RECTANGLE
DYNQ\\RECTANGLEDYNQDW.dbs"


export fluent5 \
  "C:\\FLUENT PRACTICE 3\\RECTANGLE\\RECTANGLE
DYNQ\\RECTANGLEDYNQDW.msh"


save

/-----------------------------------------------------------------/
```

B.6 VASCULARISED SOLID WITH COOLING CHANNELS

JOURNAL FILE

```
/read the dimension of the brick volume 1

/$w1 = $x2/1000 opt

/$w1 = $x2/1000

/$w1 = 0.0004

/$w1 =  $x  BF

/$w1 =  $x

/$w1 =  $x/1000  sensitivity analysis

$w1 =  $x/1000

$d1 = $w1

/$h1 = 0.01

$h1 = 0.01

/read the dimension of the brick volume 2

/$w2=$x1/1000  opt

/$w2=$x1/1000

/$w2=$dh =0.0002

/$w2=$dh BF

/$w2=$dh

/$w2=$dh/1000   sensitivity analysis

$w2=$dh/1000

$d2=$w2

/$h1 = 0.01
```

```
$h1 = 0.01

/-----------------------------------------------------------------------

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1
$offsety1 = 0.5*$d1
$offsetz1 =-0.5*$h1

/read the offset origin of brick volume 2

$offsetx2 = 0.5*$w2
$offsety2 = 0.5*$d2
$offsetz2 =-0.5*$h1

/move by  offset brick volume 2

$movex2 = 0.5*($w1-$w2)
$movey2 = 0.5*($d1-$d2)
$movez2 = 0

/-----------------------------------------------------------------------

/mesh brick V1

/$meshv1=0.09*$w1

$meshv1=0.08*$w1

/$meshv1=$mesh1

/mesh brick V2

/$meshv2=0.09*$w2

$meshv2=0.05*$w2

/$meshv2=$mesh2

/-----------------------------------------------------------------------

volume create width $w1 depth $d1 height $h1 offset $offsetx1
$offsety1 $offsetz1 brick


volume create width $w2 depth $d2 height $h1 offset $offsetx2
$offsety2 $offsetz2 brick

volume move "volume.2" offset $movex2 $movey2 $movez2

volume subtract "volume.1" volumes "volume.2" keeptool


face connect "face.7" "face.8" "face.9" "face.10" "face.11" "face.12"
\
  "face.15" "face.16" "face.17" "face.18" real
```

```
/-----------------------------------------------------------------

volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.7" "face.12" size $meshv2

/-----------------------------------------------------------------

physics create "wall front" btype "WALL" face "face.6"

physics create "wall back" btype "WALL" face "face.1"

physics create "symmetry top" btype "SYMMETRY" face "face.5"

physics create "symmetry bottom" btype "SYMMETRY" face "face.2"

physics create "symmetry left" btype "SYMMETRY" face "face.3"

physics create "symmetry right" btype "SYMMETRY" face "face.4"

physics create "wall top internal" btype "WALL" face "face.11"

physics create "wall bottom internal" btype "WALL" face "face.8"

physics create "wall left internal" btype "WALL" face "face.9"

physics create "wall right internal" btype "WALL" face "face.10"

physics create "Pressure inlet" btype "PRESSURE_INLET" face "face.12"

physics create "Pressure outlet" btype "PRESSURE_OUTLET" face

"face.7"

physics create "fluid" ctype "FLUID" volume "volume.2"

physics create "solid" ctype "SOLID" volume "volume.1"

/-----------------------------------------------------------------


save name "C:\\FLUENT PRACTICE 3\\SMART MATERIAL\\SQUARE\\SQUARESmart
DYNQ\\SquareSmartDYNQ.dbs"
export fluent5 "C:\\FLUENT PRACTICE 3\\SMART
MATERIAL\\SQUARE\\SQUARESmart DYNQ\\SquareSmartDYNQ.msh"
save

/-----------------------------------------------------------------
```

B.7 PF-1 ORIENTATION COOLING CHANNELS JOURNAL FILE

```
$w1 = $x2/1000

/$w1 = $x
```

```
$d1 = $w1

/read the dimension of the cylinder volume 2

$r1=$x1/2/1000

/$r1=$dh/2

$r3=$r1

$h1 = 0.01

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1

$offsety1 = 0.5*$d1

$offsetz1 =-0.5*$h1


/read the offset origin of cylinder volume 2

$offsetcylh1 = 0

$offsetcylr1 = 0

$offsetcylr3 =-0.5*$h1


/move by offset cylinder volume 2


$movex1 = 0.5*$w1
$movey1 = 0.5*$w1

$movez1 = 0

/-------------------------------------------------

$cmovex1 = 0
$cmovey1 = $d1

$cmovez1 = 0


/-----------------------------------------------------------------

$cmovex2 = $w1
$cmovey2 = 0
```

```
$cmovez2 = 0


/-----------------------------------------

/Optimisation   /Optimisation   /Optimisation   /Optimisation
/Optimisation

/-----------------------------------------------------------------

/mesh brick V1


/ Optimisation mesh

/$meshv1=0.15*$x2/1000  po = 0.2   0.3

$meshv1=0.15*$x2/1000

/$meshv1=0.1*$x2/1000  po = 0.1

/$meshv1=0.1*$x2/1000


/Optimisation   /Optimisation   /Optimisation   /Optimisation
/Optimisation
/-----------------------------------------------------------------


/mesh cylinder V2

/$meshv2=0.15*$x1/1000  po = 0.2   0.3

$meshv2=0.15*$x1/1000

/$meshv2=0.1*$x1/1000  po = 0.1

/$meshv2=0.1*$x1/1000

/-----------------------------------------------------------------

/normal  /normal   /normal   /normal   /normal  /normal   /normal
/normal


/$meshv1=0.15*$w1  po = 0.2

/$meshv1=0.15*$w1

/$meshv1=0.1*$w1  po = 0.1

/$meshv1=0.1*$w1

/$meshv1=0.08*$w1

/$meshv1=$mesh1
```

```
/-----------------------------------------------------------------

/normal  /normal   /normal   /normal   /normal  /normal   /normal
/normal
/-----------------------------------------------------------------

/$meshv2=0.1*2*$r1

/$meshv2=0.07*2*$r1

/$meshv2=0.15*$dh  po = 0.2


/$meshv2=0.15*$dh

/$meshv2=0.1*$dh  po = 0.1

/$meshv2=0.1*$dh

/$meshv2=$mesh2

/-----------------------------------------------------------------


volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 $offsetz1 brick

volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1

$offsetcylr1 $offsetcylr3 zaxis frustum

/-----------------------------------------------------------------

/volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 \

 / $offsetz1 brick

/volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1

$offsetcylr1 \

/  zaxis frustum


/-----------------------------------------------------------------

volume move "volume.2" offset $movex1 $movey1 $movez1


/-----------------------------------------------------------------
```

```
volume cmove "volume.1" "volume.2" multiple 1 offset $cmovex1
$cmovey1 $cmovez1
volume unite volumes "volume.1" "volume.3"
/-----------------------------------------------------------------
volume cmove "volume.1" "volume.2" "volume.4" multiple 1 offset
$cmovex2 $cmovey2 $cmovez2
volume unite volumes "volume.1" "volume.5"
/-----------------------------------------------------------------
volume subtract "volume.1" volumes "volume.2" "volume.4" "volume.6" \
  "volume.7" keeptool
/-----------------------------------------------------------------
face connect "face.1" "face.2" "face.3" "face.6" "face.7" "face.8"
"face.9" \
  "face.14" "face.16" "face.17" "face.18" "face.24" "face.25"
"face.26" \
  "face.27" "face.28" "face.29" "face.30" "face.31" "face.34"
"face.37" \
  "face.40" real
/-----------------------------------------------------------------
volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1
volume delete "volume.2" onlymesh
/-----------------------------------------------------------------
volume mesh "volume.2" cooper source "face.7" "face.9" size $meshv2
volume mesh "volume.4" cooper source "face.17" "face.18" size $meshv2
volume mesh "volume.6" cooper source "face.26" "face.27" size $meshv2
volume mesh "volume.7" cooper source "face.29" "face.30" size $meshv2
/-----------------------------------------------------------------
physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"
```

```
physics create "symmetry left" btype "SYMMETRY" face "face.3"
physics create "symmetry right" btype "SYMMETRY" face "face.24"
physics create "symmetry bottom" btype "SYMMETRY" face "face.2"
physics create "symmetry top" btype "SYMMETRY" face "face.14"
physics create "pressure inlet 1" btype "SYMMETRY" face "face.9"
physics create "pressure inlet 2" btype "SYMMETRY" face "face.18"
physics create "pressure inlet 3" btype "SYMMETRY" face "face.27"
physics create "pressure inlet 4" btype "SYMMETRY" face "face.30"
physics modify "pressure inlet 1" btype "PRESSURE_INLET" face
"face.9"
physics modify "pressure inlet 2" btype "PRESSURE_INLET" face
"face.18"
physics modify "pressure inlet 3" btype "PRESSURE_INLET" face
"face.27"
physics modify "pressure inlet 4" btype "PRESSURE_INLET" face
"face.30"
physics create "pressure outlet 1" btype "PRESSURE_OUTLET" face
"face.7"
physics create "pressure outlet 2" btype "PRESSURE_OUTLET" face
"face.17"
physics create "pressure outlet 3" btype "PRESSURE_OUTLET" face
"face.26"
physics create "pressure outlet 4" btype "PRESSURE_OUTLET" face
"face.29"
physics create "wall internal 1" btype "WALL" face "face.8"
physics create "wall internal 2" btype "WALL" face "face.16"
physics create "wall internal 3" btype "WALL" face "face.25"
physics create "wall internal 4" btype "WALL" face "face.28"
```

```
physics create "solid" ctype "SOLID" volume "volume.1"

physics create "fluid 1" ctype "FLUID" volume "volume.2"

physics create "fluid 2" ctype "FLUID" volume "volume.4"

physics create "fluid 3" ctype "FLUID" volume "volume.6"

physics create "fluid 4" ctype "FLUID" volume "volume.7"


/-----------------------------------------------------------------

save name \
  "C:\\FLUENT PRACTICE 3\\FLOW
ORIENTATION\\Parallelflow\\parallelflowchannelDYNQDW.dbs"
export fluent5 \
  "C:\\FLUENT PRACTICE 3\\FLOW
ORIENTATION\\Parallelflow\\parallelflowchannelDYNQDW.msh"
save

/-----------------------------------------------------------------
```

B.8 CF-2 ORIENTATION COOLING CHANNELS JOURNAL FILE

```
$w1 = $x2/1000

/$w1 = $x

$d1 = $w1

/read the dimension of the cylinder volume 2


$r1=$x1/2/1000

/$r1=$dh/2

$r3=$r1

$h1 = 0.01

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1

$offsety1 = 0.5*$d1

$offsetz1 =-0.5*$h1
```

```
/read the offset origin of cylinder volume 2

$offsetcylh1 = 0

$offsetcylr1 = 0

$offsetcylr3 =-0.5*$h1


/move by offset cylinder volume 2


$movex1 = 0.5*$w1

$movey1 = 0.5*$w1

$movez1 = 0


/----------------------------------------------------------------------

$cmovex1 = 0
$cmovey1 = $d1

$cmovez1 = 0
/----------------------------------------------------------------------

$cmovex2 = $w1
$cmovey2 = 0

$cmovez2 = 0

/-------------------------------------------

/mesh brick V1


/optimisation   /optimisation   /optimisation   /optimisation

/optimisation   /optimisation

/$meshv1=0.15*$x2/1000  po = 0.2    0.3

$meshv1=0.15*$x2/1000

/$meshv1=0.1*$x2/1000  po = 0.1

/$meshv1=0.1*$x2/1000

/ normal   /  normal    /  normal    /  normal   /  normal    /
normal    /  normal    /  normal   /  normal

/$meshv1=0.15*$w1  po = 0.2

/$meshv1=0.15*$w1
```

```
/$meshv1=0.1*$w1  po = 0.1

/$meshv1=0.1*$w1

/$meshv1=0.08*$w1

/$meshv1=$mesh1

/mesh cylinder V2

/optimisation   /optimisation  /optimisation   /optimisation
/optimisation

/$meshv2=0.15*$x1/1000  po = 0.2   0.3

$meshv2=0.15*$x1/1000

/$meshv2=0.1*$x1/1000  po = 0.1

/$meshv2=0.1*$x1/1000


/ normal   /  normal    /  normal    /  normal   /  normal    /
normal    /  normal    /  normal   /  normal


/$meshv2=0.1*2*$r1


/$meshv2=0.07*2*$r1

/$meshv2=0.15*$dh  po = 0.2


/$meshv2=0.15*$dh


/$meshv2=0.1*$dh  po = 0.1


/$meshv2=0.1*$dh


/$meshv2=$mesh2

/-----------------------------------------------------------------------

volume create width $w1 depth $d1 height $h1 offset $offsetx1

$offsety1 $offsetz1 brick


volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1

$offsetcylr1 $offsetcylr3 zaxis frustum
```

```
/-------------------------------------------------------------------

/volume create width $w1 depth $d1 height $h1 offset $offsetx1
$offsety1 \
 / $offsetz1 brick


/volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1
$offsetcylr1 \
/  zaxis frustum


/-------------------------------------------------------------------

volume move "volume.2" offset $movex1 $movey1 $movez1

/-------------------------------------------------------------------

volume cmove "volume.1" "volume.2" multiple 1 offset $cmovex1
$cmovey1 $cmovez1

volume unite volumes "volume.1" "volume.3"

/-------------------------------------------------------------------

volume cmove "volume.1" "volume.2" "volume.4" multiple 1 offset
$cmovex2 $cmovey2 $cmovez2

volume unite volumes "volume.1" "volume.5"

/-------------------------------------------------------------------

volume subtract "volume.1" volumes "volume.2" "volume.4" "volume.6" \
  "volume.7" keeptool

/-------------------------------------------------------------------
/-------------------------------------------------------------------

face connect "face.7" "face.8" "face.9" "face.16" "face.17" "face.18"
\
  "face.25" "face.26" "face.27" "face.28" "face.29" "face.30"
"face.1" \
  "face.2" "face.3" "face.6" "face.14" "face.24" "face.31" "face.34"
\
  "face.37" "face.40" real

/-------------------------------------------------------------------

face link "face.2" "face.14" edges "edge.5" "edge.16" vertices
"vertex.5" \
  "vertex.13" reverse periodic

/-------------------------------------------------------------------
face link "face.3" "face.24" edges "edge.10" "edge.40" vertices
"vertex.5" \
  "vertex.25" reverse periodic
```

```
/-------------------------------------------------------------------
volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1
/-------------------------------------------------------------------
volume mesh "volume.2" cooper source "face.7" "face.9" size $meshv2
volume mesh "volume.4" cooper source "face.17" "face.18" size $meshv2
volume mesh "volume.6" cooper source "face.26" "face.27" size $meshv2
volume mesh "volume.7" cooper source "face.29" "face.30" size $meshv2
/-------------------------------------------------------------------
physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"
physics create "periodic bt" btype "PERIODIC" face "face.2" "face.14"
physics create "periodic lr" btype "PERIODIC" face "face.3" "face.24"
physics create "pressure inlet f1" btype "PRESSURE_INLET" face
"face.9"
physics create "pressure inlet f3" btype "PRESSURE_INLET" face
"face.27"
physics create "pressure inlet b2" btype "PRESSURE_INLET" face
"face.17"
physics create "pressure inlet b4" btype "PRESSURE_INLET" face
"face.29"
physics create "pressure outlet b2" btype "PRESSURE_OUTLET" face
"face.7"
physics create "pressure outlet b3" btype "PRESSURE_OUTLET" face
"face.26"
physics modify "pressure outlet b2" btype label "pressure outlet b1"
face \
  "face.7"
```

```
physics create "pressure outlet f2" btype "PRESSURE_OUTLET" face
"face.18"
physics create "pressure outlet f4" btype "PRESSURE_OUTLET" face
"face.30"
physics create "wall internal 1" btype "WALL" face "face.8"
physics create "wall internal 2" btype "WALL" face "face.16"
physics create "wall internal 3" btype "WALL" face "face.25"
physics create "wall internal 4" btype "WALL" face "face.28"
/-----------------------------------------------------------------
physics create "solid" ctype "SOLID" volume "volume.1"
physics create "fluid 1" ctype "FLUID" volume "volume.2"
physics create "fluid 2" ctype "FLUID" volume "volume.4"
physics create "fluid 3" ctype "FLUID" volume "volume.6"
physics create "fluid 4" ctype "FLUID" volume "volume.7"
/-----------------------------------------------------------------
save name \
  "C:\\FLUENT PRACTICE 3\\FLOW ORIENTATION\\Counterflow
row\\CounterflowrowDYNQDW.dbs"
export fluent5 \
  "C:\\FLUENT PRACTICE 3\\FLOW ORIENTATION\\Counterflow
row\\CounterflowrowDYNQDW.msh"
save
save
/-----------------------------------------------------------------
```

B.9 CF-3 ORIENTATION COOLING CHANNELS JOURNAL FILE

```
$w1 = $x2/1000
/$w1 = $x
$d1 = $w1
/read the dimension of the cylinder volume 2
$r1=$x1/2/1000
/$r1=$dh/2
```

```
$r3=$r1

$h1 = 0.01

/-----------------------------------------------------------------

/read the offset origin of brick volume 1

$offsetx1 = 0.5*$w1

$offsety1 = 0.5*$d1

$offsetz1 =-0.5*$h1

/read the offset origin of cylinder volume 2

$offsetcylh1 = 0

$offsetcylr1 = 0

$offsetcylr3 =-0.5*$h1

/move by offset cylinder volume 2

$movex1 = 0.5*$w1

$movey1 = 0.5*$w1

$movez1 = 0

/-------------------------------------------------

$cmovex1 = 0

$cmovey1 = $d1

$cmovez1 = 0

/-----------------------------------------------------------------

$cmovex2 = $w1

$cmovey2 = 0

$cmovez2 = 0

/-----------------------------------------------------------------

/mesh brick V1

/Optimisation /Optimisation /Optimisation /Optimisation /Optimisation

/Optimisation /Optimisation

/$meshv1=0.15*$x2/1000  po = 0.2   0.3

$meshv1=0.15*$x2/1000
```

```
/$meshv1=0.1*$x2/1000  po = 0.1
/$meshv1=0.1*$x2/1000
/Normal /Normal  /Normal  /Normal  /Normal  /Normal  /Normal  /Normal
/Normal  /Normal  /Normal  /Normal
/$meshv1=0.15*$w1  po = 0.2
/$meshv1=0.15*$w1
/$meshv1=0.1*$w1  po = 0.1
/$meshv1=0.16*$w1
/$meshv1=0.08*$w1
/$meshv1=$mesh1
/mesh cylinder V2
/Optimisation /Optimisation /Optimisation /Optimisation /Optimisation
/Optimisation /Optimisation
/$meshv2=0.15*$x1/1000  po = 0.2   0.3
$meshv2=0.15*$x1/1000
/$meshv2=0.1*$x1/1000  po = 0.1
/$meshv2=0.1*$x1/1000
/Normal /Normal  /Normal  /Normal  /Normal  /Normal  /Normal  /Normal
/Normal  /Normal  /Normal  /Normal
/$meshv2=0.1*2*$r1
/$meshv2=0.07*2*$r1
/$meshv2=0.15*$dh  po = 0.2
/$meshv2=0.15*$dh
/$meshv2=0.15*$dh  po = 0.1
/$meshv2=0.1*$dh
/$meshv2=$mesh2
/-----------------------------------------------------------------
volume  create  width  $w1  depth  $d1  height  $h1  offset  $offsetx1
$offsety1 $offsetz1 brick
```

```
volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1
$offsetcylr1 $offsetcylr3 zaxis frustum
/-----------------------------------------------------------------
/volume create width $w1 depth $d1 height $h1 offset $offsetx1
$offsety1 \
 / $offsetz1 brick
/volume create height $h1 radius1 $r1 radius3 $r3 offset $offsetcylh1
$offsetcylr1 \
/  zaxis frustum
/-----------------------------------------------------------------
volume move "volume.2" offset $movex1 $movey1 $movez1
/-----------------------------------------------------------------
volume cmove "volume.1" "volume.2" multiple 1 offset $cmovex1
$cmovey1 $cmovez1
volume unite volumes "volume.1" "volume.3"
/-----------------------------------------------------------------
volume cmove "volume.1" "volume.2" "volume.4" multiple 1 offset
$cmovex2 $cmovey2 $cmovez2
volume unite volumes "volume.1" "volume.5"
/-----------------------------------------------------------------
volume subtract "volume.1" volumes "volume.2" "volume.4" "volume.6" \
  "volume.7" keeptool
/-----------------------------------------------------------------
face connect "face.1" "face.2" "face.3" "face.6" "face.7" "face.8"
"face.9" \
  "face.14" "face.16" "face.17" "face.18" "face.24" "face.25"
"face.26" \
  "face.27" "face.28" "face.29" "face.30" "face.31" "face.34"
"face.37" \
```

```
  "face.40" real
/-----------------------------------------------------------------
face link "face.2" "face.14" edges "edge.5" "edge.16" vertices
"vertex.5" \
  "vertex.13" reverse periodic
/-----------------------------------------------------------------
face link "face.3" "face.24" edges "edge.10" "edge.40" vertices
"vertex.5" \
  "vertex.25" reverse periodic
/-----------------------------------------------------------------
volume mesh "volume.1" cooper source "face.1" "face.6" size $meshv1

volume mesh "volume.2" cooper source "face.7" "face.9" size $meshv2
volume mesh "volume.4" cooper source "face.17" "face.18" size $meshv2
volume mesh "volume.6" cooper source "face.26" "face.27" size $meshv2
volume mesh "volume.7" cooper source "face.29" "face.30" size $meshv2
/-----------------------------------------------------------------
physics create "wall front" btype "WALL" face "face.6"
physics create "wall back" btype "WALL" face "face.1"
physics create "periodic lr" btype "PERIODIC" face "face.3" "face.24"
physics create "periodic bt" btype "PERIODIC" face "face.2" "face.14"
physics create "pressure inlet f1" btype "PRESSURE_INLET" face
"face.9"
physics create "pressure inlet b2" btype "PRESSURE_INLET" face
"face.17"
physics create "pressure inlet b3" btype "PRESSURE_INLET" face
"face.26"
physics create "pressure inlet f4" btype "PRESSURE_INLET" face
"face.30"
```

```
physics create "pressure outlet b1" btype "PRESSURE_OUTLET" face
"face.7"
physics create "pressure outlet f2" btype "PRESSURE_OUTLET" face
"face.18"
physics create "pressure outlet f3" btype "PRESSURE_OUTLET" face
"face.27"
physics create "pressure outlet b4" btype "PRESSURE_OUTLET" face
"face.29"
physics create "wall internal 1" btype "WALL" face "face.8"
physics create "wall internal 2" btype "WALL" face "face.16"
physics create "wall internal 3" btype "WALL" face "face.25"
physics create "wall internal 4" btype "WALL" face "face.28"
physics create "solid" ctype "SOLID" volume "volume.1"
physics create "fluid 1" ctype "FLUID" volume "volume.2"
physics create "fluid 2" ctype "FLUID" volume "volume.4"
physics create "fluid 3" ctype "FLUID" volume "volume.6"
physics create "fluid 4" ctype "FLUID" volume "volume.7"
/-----------------------------------------------------------------
save name \
  "C:\\FLUENT      PRACTICE      3\\FLOW      ORIENTATION\\Counterflow
channels\\CounterflowchannelsDYNQDW.dbs"
export fluent5 \
  "C:\\FLUENT      PRACTICE      3\\FLOW      ORIENTATION\\Counterflow
channels\\CounterflowchannelsDYNQDW.msh"
save
/-----------------------------------------------------------------
```

C

APPENDIX C: FLUENT JOURNAL FILEN FOR NUMERICAL SIMULATION OF COOLING CHANNELS

C.1 COOLING CHANNELS FLUENT JOURNAL FILE

```
FLUENT
Version: 3d, dp, pbns, lam (3d, double precision, pressure-based,
laminar)
Release: 13.0.0
Title:

Models
------

   Model                         Settings
   ------------------------------------
   Space                         3D
   Time                          Steady
   Viscous                       Laminar
   Heat Transfer                 Enabled
   Solidification and Melting    Disabled
   Radiation                     None
   Species                       Disabled
   Coupled Dispersed Phase       Disabled
   NOx Pollutants                Disabled
   SOx Pollutants                Disabled
   Soot                          Disabled
   Mercury Pollutants            Disabled



Material Properties
-------------------

   Material: silicon (solid)

      Property                Units    Method     Value(s)
      ------------------------------------------------------
      Density                 kg/m3    constant   2330
      Cp (Specific Heat)      j/kg-k   constant   720
```

```
      Thermal Conductivity   w/m-k    constant   148

   Material: water-liquid (fluid)

      Property                          Units      Method     Value(s)
      ------------------------------------------------------------------
-
      Density                           kg/m3      constant   998.20001
      Cp (Specific Heat)                j/kg-k     constant   4182
      Thermal Conductivity              w/m-k      constant   0.60000002
      Viscosity                         kg/m-s     constant   0.001003
      Molecular Weight                  kg/kgmol   constant   18.0152
      Thermal Expansion Coefficient     1/k        constant   0
      Speed of Sound                    m/s        none       #f

   Material: air (fluid)

      Property                          Units      Method     Value(s)
      ------------------------------------------------------------------
-
      Density                           kg/m3      constant   1.225
      Cp (Specific Heat)                j/kg-k     constant   1006.43
      Thermal Conductivity              w/m-k      constant   0.0242
      Viscosity                         kg/m-s     constant   1.7894e-
05
      Molecular Weight                  kg/kgmol   constant   28.966
      Thermal Expansion Coefficient     1/k        constant   0
      Speed of Sound                    m/s        none       #f

   Material: aluminum (solid)

      Property               Units    Method     Value(s)
      ------------------------------------------------------
      Density                kg/m3    constant   2719
      Cp (Specific Heat)     j/kg-k   constant   871
      Thermal Conductivity   w/m-k    constant   202.4

Cell Zone Conditions
--------------------

   Zones

      name    id   type
      ------------------
      fluid   2    fluid
      solid   3    solid

   Setup Conditions

      fluid

         Condition                                                  Value
         ------------------------------------------------------------------
------------------------------------------------------------------------
------------------------------------------------------------------------
```

```
------------------------------------------------------------------------
-----------------
         Material Name
water-liquid
         Specify source terms?                                      no
         Source Terms
((mass) (x-momentum) (y-momentum) (z-momentum) (energy))
         Specify fixed values?                                      no
         Local Coordinate System for Fixed Velocities               no
         Fixed Values
((x-velocity (inactive . #f) (constant . 0) (profile  )) (y-velocity
(inactive . #f) (constant . 0) (profile  )) (z-velocity (inactive .
#f) (constant . 0) (profile  )) (temperature (inactive . #f)
(constant . 0) (profile  )))
         Frame Motion?                                              no
         Relative To Cell Zone                                      -1
         Reference Frame Rotation Speed (rad/s)                      0
         Reference Frame X-Velocity Of Zone (m/s)                    0
         Reference Frame Y-Velocity Of Zone (m/s)                    0
         Reference Frame Z-Velocity Of Zone (m/s)                    0
         Reference Frame X-Origin of Rotation-Axis (m)               0
         Reference Frame Y-Origin of Rotation-Axis (m)               0
         Reference Frame Z-Origin of Rotation-Axis (m)               0
         Reference Frame X-Component of Rotation-Axis                0
         Reference Frame Y-Component of Rotation-Axis                0
         Reference Frame Z-Component of Rotation-Axis                1
         Reference Frame User Defined Zone Motion Function        none
         Mesh Motion?                                               no
         Relative To Cell Zone                                      -1
         Moving Mesh Rotation Speed (rad/s)                          0
         Moving Mesh X-Velocity Of Zone (m/s)                        0
         Moving Mesh Y-Velocity Of Zone (m/s)                        0
         Moving Mesh Z-Velocity Of Zone (m/s)                        0
         Moving Mesh X-Origin of Rotation-Axis (m)                   0
         Moving Mesh Y-Origin of Rotation-Axis (m)                   0
         Moving Mesh Z-Origin of Rotation-Axis (m)                   0
         Moving Mesh X-Component of Rotation-Axis                    0
         Moving Mesh Y-Component of Rotation-Axis                    0
         Moving Mesh Z-Component of Rotation-Axis                    1
         Moving Mesh User Defined Zone Motion Function            none
         Deactivated Thread                                         no
         Embedded Subgrid-Scale Model                                0
         Momentum Spatial Discretization                             0
         Cwale                                                   0.325
         Cs                                                        0.1
         Porous zone?                                               no
         Conical porous zone?                                       no
         X-Component of Direction-1 Vector                           1
         Y-Component of Direction-1 Vector                           0
         Z-Component of Direction-1 Vector                           0
         X-Component of Direction-2 Vector                           0
         Y-Component of Direction-2 Vector                           1
         Z-Component of Direction-2 Vector                           0
         X-Component of Cone Axis Vector                             1
         Y-Component of Cone Axis Vector                             0
         Z-Component of Cone Axis Vector                             0
```

```
         X-Coordinate of Point on Cone Axis (m)                    1

         Y-Coordinate of Point on Cone Axis (m)                    0
         Z-Coordinate of Point on Cone Axis (m)                    0
         Half Angle of Cone Relative to its Axis (deg)             0
         Relative Velocity Resistance Formulation?               yes
         Direction-1 Viscous Resistance (1/m2)                     0
         Direction-2 Viscous Resistance (1/m2)                     0
         Direction-3 Viscous Resistance (1/m2)                     0
         Choose alternative formulation for inertial resistance?   no
         Direction-1 Inertial Resistance (1/m)                     0
         Direction-2 Inertial Resistance (1/m)                     0
         Direction-3 Inertial Resistance (1/m)                     0
         C0 Coefficient for Power-Law                              0
         C1 Coefficient for Power-Law                              0
         Porosity                                                  1
         Solid Material Name                                 aluminum

      solid

         Condition                                                 Value
         ------------------------------------------------------------
------------------------------------------------
         Material Name                                          silicon
         Specify source terms?                                  yes
         Source Terms                                           ((energy
((constant . 1e+08) (inactive . #f) (profile  ))))
         Specify fixed values?                                  no
         Fixed Values
((temperature (inactive . #f) (constant . 0) (profile  )))
         Frame Motion?                                          no
         Relative To Cell Zone                                  -1
         Reference Frame Rotation Speed (rad/s)                 0
         Reference Frame X-Velocity Of Zone (m/s)               0
         Reference Frame Y-Velocity Of Zone (m/s)               0
         Reference Frame Z-Velocity Of Zone (m/s)               0
         Reference Frame X-Origin of Rotation-Axis (m)          0
         Reference Frame Y-Origin of Rotation-Axis (m)          0
         Reference Frame Z-Origin of Rotation-Axis (m)          0
         Reference Frame X-Component of Rotation-Axis           0
         Reference Frame Y-Component of Rotation-Axis           0
         Reference Frame Z-Component of Rotation-Axis           1
         Reference Frame User Defined Zone Motion Function      none
         Mesh Motion?                                           no
         Relative To Cell Zone                                  -1
         Moving Mesh Rotation Speed (rad/s)                     0
         Moving Mesh X-Velocity Of Zone (m/s)                   0
         Moving Mesh Y-Velocity Of Zone (m/s)                   0
         Moving Mesh Z-Velocity Of Zone (m/s)                   0
         Moving Mesh X-Origin of Rotation-Axis (m)              0
         Moving Mesh Y-Origin of Rotation-Axis (m)              0
         Moving Mesh Z-Origin of Rotation-Axis (m)              0
         Moving Mesh X-Component of Rotation-Axis               0
         Moving Mesh Y-Component of Rotation-Axis               0
         Moving Mesh Z-Component of Rotation-Axis               1
         Moving Mesh User Defined Zone Motion Function          none
         Deactivated Thread                                     no
```

```
Boundary Conditions

--------------------

   Zones

      name                      id   type
      ----------------------------------------
      wall_internal-shadow      15   wall
      wall_internal             4    wall
      pressure_outlet           5    pressure-outlet
      pressure_inlet            6    pressure-inlet
      symmetry_right            7    symmetry
      symmetry_left             8    symmetry
      symmetry_bottom           9    symmetry
      symmetry_top              10   symmetry
      wall_back                 11   wall
      wall_front                12   wall

   Setup Conditions

      wall_internal-shadow

         Condition                                                  Value
         ----------------------------------------------------------------
         Wall Thickness (m)                                       0
         Heat Generation Rate (w/m3)                              0
         Material Name                                            silicon
         Thermal BC Type                                          3
         Temperature (c)                                          26.85
         Heat Flux (w/m2)                                         0
         Convective Heat Transfer Coefficient (w/m2-k)            0
         Free Stream Temperature (c)                              26.85
         Enable shell conduction?                                 no
         Wall Motion                                              0
         Shear Boundary Condition                                 0
         Define wall motion relative to adjacent cell zone?       yes
         Apply a rotational velocity to this wall?                no
         Velocity Magnitude (m/s)                                 0
         X-Component of Wall Translation                          1
         Y-Component of Wall Translation                          0
         Z-Component of Wall Translation                          0
         Define wall velocity components?                         no
         X-Component of Wall Translation (m/s)                    0
         Y-Component of Wall Translation (m/s)                    0
         Z-Component of Wall Translation (m/s)                    0
         External Emissivity                                      1
         External Radiation Temperature (c)                       26.85
         Rotation Speed (rad/s)                                   0
         X-Position of Rotation-Axis Origin (m)                   0
         Y-Position of Rotation-Axis Origin (m)                   0
         Z-Position of Rotation-Axis Origin (m)                   0
         X-Component of Rotation-Axis Direction                   0
         Y-Component of Rotation-Axis Direction                   0
         Z-Component of Rotation-Axis Direction                   1
         X-component of shear stress (pascal)                     0
```

```
   Y-component of shear stress (pascal)                  0
   Z-component of shear stress (pascal)                  0
   Surface tension gradient (n/m-k)                      0

Specularity Coefficient                                  0

wall_internal

   Condition                                             Value
   ------------------------------------------------------------
   Wall Thickness (m)                                    0
   Heat Generation Rate (w/m3)                           0
   Material Name                                         silicon
   Thermal BC Type                                       3
   Temperature (c)                                       26.85
   Heat Flux (w/m2)                                      0
   Convective Heat Transfer Coefficient (w/m2-k)         0
   Free Stream Temperature (c)                           26.85
   Enable shell conduction?                              no
   Wall Motion                                           0
   Shear Boundary Condition                              0
   Define wall motion relative to adjacent cell zone?    yes
   Apply a rotational velocity to this wall?             no
   Velocity Magnitude (m/s)                              0
   X-Component of Wall Translation                       1
   Y-Component of Wall Translation                       0
   Z-Component of Wall Translation                       0
   Define wall velocity components?                      no
   X-Component of Wall Translation (m/s)                 0
   Y-Component of Wall Translation (m/s)                 0
   Z-Component of Wall Translation (m/s)                 0
   External Emissivity                                   1
   External Radiation Temperature (c)                    26.85
   Rotation Speed (rad/s)                                0
   X-Position of Rotation-Axis Origin (m)                0
   Y-Position of Rotation-Axis Origin (m)                0
   Z-Position of Rotation-Axis Origin (m)                0
   X-Component of Rotation-Axis Direction                0
   Y-Component of Rotation-Axis Direction                0
   Z-Component of Rotation-Axis Direction                1
   X-component of shear stress (pascal)                  0
   Y-component of shear stress (pascal)                  0
   Z-component of shear stress (pascal)                  0
   Surface tension gradient (n/m-k)                      0
   Specularity Coefficient                               0

pressure_outlet

   Condition                                             Value
   ------------------------------------------------------------
   Gauge Pressure (pascal)                               0
   Backflow Total Temperature (c)                        26.85
   Backflow Direction Specification Method               1
   Coordinate System                                     0
   X-Component of Flow Direction                         1
   Y-Component of Flow Direction                         0
   Z-Component of Flow Direction                         0
```

```
   X-Component of Axis Direction                     1
   Y-Component of Axis Direction                     0
   Z-Component of Axis Direction                     0
   X-Coordinate of Axis Origin (m)                   0

   Y-Coordinate of Axis Origin (m)                   0
   Z-Coordinate of Axis Origin (m)                   0
   is zone used in mixing-plane model?               no
   Radial Equilibrium Pressure Distribution          no
   Specify Average Pressure Specification            no
   Specify targeted mass flow rate                   no
   Targeted mass flow (kg/s)                         1
   Upper Limit of Absolute Pressure Value (pascal)   5000000
   Lower Limit of Absolute Pressure Value (pascal)   1

pressure_inlet

   Condition                                         Value
   ------------------------------------------------------
   Reference Frame                                   0
   Gauge Total Pressure (pascal)                     5000
   Supersonic/Initial Gauge Pressure (pascal)        0
   Total Temperature (c)                             26.85
   Direction Specification Method                    1
   Coordinate System                                 0
   X-Component of Flow Direction                     1
   Y-Component of Flow Direction                     0
   Z-Component of Flow Direction                     0
   X-Component of Flow Direction                     1
   Y-Component of Flow Direction                     0
   Z-Velocity (m/s)                                  0
   X-Component of Axis Direction                     1
   Y-Component of Axis Direction                     0
   Z-Component of Axis Direction                     0
   X-Coordinate of Axis Origin (m)                   0
   Y-Coordinate of Axis Origin (m)                   0
   Z-Coordinate of Axis Origin (m)                   0
   is zone used in mixing-plane model?               no

symmetry_right

   Condition   Value
   -----------------

symmetry_left

   Condition   Value
   -----------------

symmetry_bottom

   Condition   Value
   -----------------

symmetry_top

   Condition   Value
```

```
   -----------------

wall_back

   Condition                                                Value

   ---------------------------------------------------------------
   Wall Thickness (m)                                       0
   Heat Generation Rate (w/m3)                              0
   Material Name                                            silicon
   Thermal BC Type                                          1
   Temperature (c)                                          26.85
   Heat Flux (w/m2)                                         0
   Convective Heat Transfer Coefficient (w/m2-k)            0
   Free Stream Temperature (c)                              26.85
   Enable shell conduction?                                 no
   Wall Motion                                              0
   Shear Boundary Condition                                 0
   Define wall motion relative to adjacent cell zone?       yes
   Apply a rotational velocity to this wall?                no
   Velocity Magnitude (m/s)                                 0
   X-Component of Wall Translation                          1
   Y-Component of Wall Translation                          0
   Z-Component of Wall Translation                          0
   Define wall velocity components?                         no
   X-Component of Wall Translation (m/s)                    0
   Y-Component of Wall Translation (m/s)                    0
   Z-Component of Wall Translation (m/s)                    0
   External Emissivity                                      1
   External Radiation Temperature (c)                       26.85
   Rotation Speed (rad/s)                                   0
   X-Position of Rotation-Axis Origin (m)                   0
   Y-Position of Rotation-Axis Origin (m)                   0
   Z-Position of Rotation-Axis Origin (m)                   0
   X-Component of Rotation-Axis Direction                   0
   Y-Component of Rotation-Axis Direction                   0
   Z-Component of Rotation-Axis Direction                   1
   X-component of shear stress (pascal)                     0
   Y-component of shear stress (pascal)                     0
   Z-component of shear stress (pascal)                     0
   Surface tension gradient (n/m-k)                         0
   Specularity Coefficient                                  0

wall_front

   Condition                                                Value
   ---------------------------------------------------------------
   Wall Thickness (m)                                       0
   Heat Generation Rate (w/m3)                              0
   Material Name                                            silicon
   Thermal BC Type                                          1
   Temperature (c)                                          26.85
   Heat Flux (w/m2)                                         0
   Convective Heat Transfer Coefficient (w/m2-k)            0
   Free Stream Temperature (c)                              26.85
   Enable shell conduction?                                 no
   Wall Motion                                              0
```

```
          Shear Boundary Condition                              0
          Define wall motion relative to adjacent cell zone?   yes
          Apply a rotational velocity to this wall?            no
          Velocity Magnitude (m/s)                             0
          X-Component of Wall Translation                      1
          Y-Component of Wall Translation                      0

          Z-Component of Wall Translation                      0
          Define wall velocity components?                     no
          X-Component of Wall Translation (m/s)                0
          Y-Component of Wall Translation (m/s)                0
          Z-Component of Wall Translation (m/s)                0
          External Emissivity                                  1
          External Radiation Temperature (c)                   26.85
          Rotation Speed (rad/s)                               0
          X-Position of Rotation-Axis Origin (m)               0
          Y-Position of Rotation-Axis Origin (m)               0
          Z-Position of Rotation-Axis Origin (m)               0
          X-Component of Rotation-Axis Direction               0
          Y-Component of Rotation-Axis Direction               0
          Z-Component of Rotation-Axis Direction               1
          X-component of shear stress (pascal)                 0
          Y-component of shear stress (pascal)                 0
          Z-component of shear stress (pascal)                 0
          Surface tension gradient (n/m-k)                     0
          Specularity Coefficient                              0

Solver Settings
---------------

   Equations

      Equation   Solved
      -----------------
      Flow       yes
      Energy     yes

   Numerics

      Numeric                            Enabled
      ---------------------------------------
      Absolute Velocity Formulation      yes

   Relaxation

      Variable      Relaxation Factor
      -------------------------------
      Pressure      0.30000001
      Density       1
      Body Forces   1
      Momentum      0.69999999
      Energy        1

   Linear Solver

                    Solver     Termination   Residual Reduction
      Variable      Type       Criterion     Tolerance
```

```
      ------------------------------------------------------
      Pressure     V-Cycle    0.1
      X-Momentum   Flexible   0.1           0.7
      Y-Momentum   Flexible   0.1           0.7
      Z-Momentum   Flexible   0.1           0.7
      Energy       Flexible   0.1           0.7

   Pressure-Velocity Coupling

      Parameter   Value
      ------------------
      Type        SIMPLE

   Discretization Scheme

      Variable   Scheme
      ------------------------------
      Pressure   Second Order
      Momentum   Second Order Upwind
      Energy     Second Order Upwind

   Solution Limits

      Quantity                     Limit
      ---------------------------------
      Minimum Absolute Pressure    1
      Maximum Absolute Pressure    5e+10
      Minimum Temperature          1
      Maximum Temperature          5000

;;/-------------------------------------------------------------------/;;
```

C.2 COOLING CHANNELS FLUENT BOUNDARY CONDITIONS JOURNAL FILE

```
;;1-------------------------------------------------------------------
-------------------------

;;cylinder50kbc

;; Read Mesh and Scale Mesh

file/set-batch-options no yes no

file/start-transcript CylinderDYNQDW_trans.trn

file/read-case CylinderDYNQDW.msh

grid/scale 1 1 1
```

```
;; Read Boundary Conditions

file/read-bc cylinder50kbc

;;file/read-bc CylinderBe=10^3

;; Define Models and Units

define/models/energy yes no no no yes

define/models/viscous/laminar yes

define/units temperature c

;; Monitors

solve/monitors/residual/plot yes

solve/monitors/residual/print yes

solve/monitors/residual/convergence-criteria 1e-6 1e-6 1e-6 1e-6 1e-
10

;; Initialize and Solve

solve/initialize/compute-defaults all-zones

solve/initialize/initialize-flow

solve/iterate 300

;; Post Processing

file/export/ascii CylinderTemp.dat default-interior:001 default-
interior wall_front

wall_back symmetry_top symmetry_bottom symmetry_left symmetry_right

pressure_inlet pressure_outlet wall_internal wall_internal-shadow ()
no temperature ()

no

file/stop-transcript

file/write-case-data CylinderDYNQDW_data.cas.gz


exit

/-----------------------------------------------------
```