Pontificia Universidad JAVERIANA Bogotá

Univ.Sci. 26(2): 139–157, 2021
doi:10.11144/Javeriana.SC26-2.eodt

SUPPLEMENTARY MATERIAL 1

Extension of Dasgupta's Technique for Higher Degree Approximation

P. L. Powar[1], **Rishabh Tiwari**[1], **Vishnu Narayan Mishra***[2]

Mathematica program for degree two approximation

```
n = 5;
(* Nodal points *)
xy = {{0, 1, 7/5, 3/5, 0}, {0, 0, 3/5, 7/5, 1}}
uv = {{1/2, 12/10, 1, 3/10, 0}, {0, 3/10, 1, 12/10, 1/2}}
node = Transpose[xy]
(*Intermediate nodes*)
nod = Transpose[uv]
(*Linear forms*)
Intercept[{x1_,y1_},{x2_,y2_}] := {y2-y1, x1-x2, -x1*y2 + y1*x2}
nodesToIntercepts[p_?MatrixQ] := Thread[Intercept[p,RotateRight[p]]]
nodesToLines[p_, {x_,y_}] := -Dot[nodesToIntercepts[p], {x,y,1}]
lst = nodesToLines[nod, {x, y}];
ls = nodesToLines[node, {x, y}];
(* Formation of Degree two wedge functions *)
k = {k1, k2, k3, k4, k5, k6, k7, k8, k9, k10};
For[i = 1, i < 6, i++, m = Mod[i+2, 5];
 If[m == 0, num[i] = k[[i]]*ls[[i+2]]*ls[[1]]*ls[[2]]*lst[[i]],
  If[Mod[m + 2, 5] == 0,
   num[i] = k[[i]]*ls[[m]]*ls[[m+1]]*ls[[5]]*lst[[i]],
   num[i] = k[[i]]*ls[[m]]*ls[[m+1]]*ls[[Mod[m+2, 5]]]*lst[[i]]
  ]
 ]
];
For[j = 6, j < 11, j++,
 num[j] = (k[[j]]*num[j-5]*ls[[j-5]])/(k[[j-5]]*lst[[j-5]]);
]
deno = Sum[num[i], {i, 10}]
L = Coefficient[deno, x, 4], Coefficient[deno, x^3*y, 1],
    Coefficient[deno, x^2*y^2, 1], Coefficient[deno, x*y^2, 1],
    Coefficient[deno, x^2*y, 1], Coefficient[deno, y^3, 1],
    Coefficient[deno, xy^3, 1], Coefficient[deno, y^4, 1],
    Coefficient[deno, x^3, 1]};
k1 := 1
Table[L[[i]] == 0, {i, 9}];
Solve[%, {k2, k3, k4, k5, k6, k7, k8, k9, k10}]
For[i = 1, i < 11,  i++,
(Subscript[N, i]^2)[{x, y}] = Thread[Expand[num[i]]/deno] /. %]
(* Unknown Function f, to be approximated *)
ClearAll[f, x, y, i, j];
f[{x_, y_}] = Sin[x*y]
(* Coefficients *)
For[i = 1, i < 11, i++,
```

```
 If[i < 6, a[i] = f[node[[i]]],
 If[Mod[i,5] == 0, a[i] = f[nod[[5]]], a[i] = f[nod[[Mod[i,5]]]]]
 ]
]
ClearAll[A1];
(* Approximation *)
A1 = Thread[Apart[Sum[a[i]*(Subscript[N,i]^2)[{x,y}], {i,1,10}]]]
(* Plot *)
Plot3D[A1, {x, y} ∈ Polygon[node]]
g[{x_, y_}] = Simplify[Abs[A1 - f[{x, y}]]]
NIntegrate[g[{x, y}], {x, y} ∈ Polygon[node]]
Plot3D[g[{x, y}], {x, y} ∈ Polygon[node],
ColorFunction ⟶ Function[{x, y, z}, Hue[z]]]
```