مجله علمی-پژوهشی
رایانش نرم و فناوری اطلاعات
جلد۱، شماره۱، سال ۱۳۹۱

اصلاح مشکلات توابع برازش الگوریتم‌های تکاملی در شبکه‌ی حسگر بی‌سیم

عطاالله ابراهیم زاده[۱]،عطیه حسین‌پور[۲]،مجتبی جعفری لاهیجانی[۳]، مریم نجیمی [۴]، محمدحسین پور [۵]
[۱]دانشگاه صنعتی نوشیروانی بابل، E_Zadeh@nit.ac.ir
[۲]دانشگاه علوم و فنون مازندران، A.Hoseinpour@ustmb.ac.ir
[۳]دانشگاه علوم و فنون مازندران، Jafari@ustmb.ac.ir
[۴] دانشگاه صنعتی نوشیروانی بابل، M.Najimi@stu.nit.ac.ir
دانشگاه صنعتی شریف، Hoseinpour.mhmd@gmail.com

چکیده:
شبکه‌های حسگر بی‌سیم علاوه بر جذابیت و کاربرد فراوان، دارای چالش‌هایی نظیر مصرف انرژی، مکان‌یابی، امنیت و غیره می‌باشند که عمدتا برای حل آنها به سراغ الگوریتم‌های تکاملی می‌روند. یکی از مهم‌ترین اجزاء هر الگوریتم تکاملی، تابع برازش آن می‌باشد که در واقع کل مسئله را به سمت بهینه شدن هدایت می‌کند. توابع برازش مورد استفاده در شبکه‌های حسگر بی‌سیم اکثرا دارای مشکلاتی نظیر عدم توجه به واحد پارامترها هنگام کنار هم قرار دادن آنها تحت یک رابطه‌ی ریاضی و عدم توجه به وابستگی تاثیر تغییرات یک پارامتر به سایر پارامترها است. در این مقاله به بررسی و رفع مشکلات توابع برازش خواهیم پرداخت.

کلید واژه‌ها: *شبکه‌ی حسگر بی‌سیم، تابع برازش، الگوریتم تکاملی*

۱. مقدمه

امروزه زندگی بدون ارتباطات بی‌سیم قابل تصور نیست. پیشرفت تکنولوژی و ایجاد مدارات کوچک و کوچکتر باعث شده است تا استفاده از مدارات بی‌سیم در اغلب وسایل الکترونیکی امروز ممکن شود. این پیشرفت همچنین باعث توسعه ریز-حسگرها نیز شده است. این ریز حسگرها توانایی‌های زیادی در امر نظارت بر محیط اطراف دارند از جمله کاربردهای آنان می‌توان به شناسایی صدا برای حس کردن زلزله و جمع‌آوری اطلاعات در مناطق دور افتاده و همچنین مکان‌هایی که برای اکتشافات انسانی مناسب نیستند اشاره کرد [3][2][1]. این شبکه‌ها علاوه بر جذابیت و کاربرد فراوان، دارای چالش‌هایی نیز می‌باشند. از جمله این چالش‌ها می‌توان به معضل مصرف انرژی، مکان‌یابی، امنیت شبکه‌ی حسگر، و غیره اشاره کرد که عمدتا برای حل آنها به سراغ روش‌های نرم‌افزاری، نظیر الگوریتم‌های تکاملی می‌روند. همان‌طور که می‌دانید یکی از مهم‌ترین اجزاء هر الگوریتم تکاملی، تابع برازش آن می‌باشد که در واقع کل مسئله را به سمت بهینه شدن هدایت می‌کند. از این رو انتخاب پارامترهای مناسب با هدف مسئله و نحوه‌ی استفاده از این پارامترها تحت یک رابطه‌ی ریاضی از اهمیت بالایی برخوردار است. تابع برازش مورد استفاده در اکثر مقالاتی که در ارتباط با کاربرد الگوریتم‌های تکاملی در زمینه‌های مختلف شبکه‌ی حسگر بی‌سیم هستند دارای مشکلاتی نظیر عدم توجه به واحد پارامترها هنگام کنار هم قرار دادن آنها تحت یک

رابطه‌ی ریاضی و عدم توجه به وابستگی تاثیر تغییراتیک پارامتر به سایر پارامترها است. در این مقاله به بررسی و رفع مشکلات توابع برازش، پرداخته خواهد شد.

این مقاله در ادامه به صورت زیر سازماندهی شده است:

در بخش دوم، مشکلات توابع برازش بیان شده و به بررسی چند تابع برازش که در مقالات مختلف آمده است، پرداخته می‌شود. در بخش سوم، راه‌حل‌هایی برای مشکلات بیان شده، ارائه خواهد شد. و در بخش چهار، نتیجه‌ای به صورت خلاصه‌وار بیان می‌شود.

۲. بررسی توابع برازش

برای بهینه‌سازی هر مسئله‌ای در کنار روند کلی الگوریتم نیاز به یک تابع برازش می‌باشد. به دلیل این که این تابع دارای اهمیت بسیار زیادی در بهینه‌سازی مسئله، یعنی پیدا نمودن بهترین جواب و سرعت بخشیدن در روند الگوریتم دارد، پس باید دقت بیشتری در ساخت این تابع نمود. با مطالعه‌ی مقالات متعدد در زمینه‌های مختلف کاربرد الگوریتم‌های تکاملی همانند الگوریتم ژنتیک، زنبورها، ذرات و غیره در شبکه‌های حسگر بی‌سیم متوجه‌ی مشکلاتی در توابع برازش آنها شدیم. که این مشکلات در ادامه‌ی این بخش بررسی خواهند شد.

۲,۱ بیان مشکلات توابع برازش

در تمامی مقالات [8][7][6][5][4] هنگام ساخت تابع برازش به دو نکته‌ی بسیار مهم توجه نشده است. ۱) یکسان نبودن واحد و درجه‌ی پارامترهایی که در کنار هم و در یک رابطه قرار می‌گیرند. ۲) استفاده از عملگر جمع که وابستگی تأثیر تغییرات یک پارامتر به سایر پارامترها را در نظر نمی‌گیرد.

برای درک بهتر موارد مذکور، به مثال زیر توجه کنید:

مثال) فرض کنید هنگام پیاده‌سازی یک الگوریتم تکاملی که دارای دو پارامتر E(انرژی) و d(فاصله) می‌باشد، در مراحل مختلف مقادیر زیر را برای تابع برازش داشته باشیم:

		E (kgm^2/s^2)		d (m)	
F_1	=	2.25×10^5	+	2×10^1	= 225020
F_2	=	3×10^5	+	3×10^1	= 300030
F_3	=	2×10^5	+	4×10^1	= 200040
F_4	=	1.5×10^5	+	5×10^1	= 150020
F_5	=	4.25×10^5	+	0	= 425000

مشکلاتی که در این مثال وجود دارد:

۱) هر کدام از پارامترها دارای واحدهایی متفاوت می‌باشند (E براساس kgm^2/s^2 و d براساس m). در این صورت نمی‌توانیم این پارامترها را در کنار هم و تحت یک رابطه‌ی ریاضی قرار دهیم.

۲) همان‌طور که مشاهده می‌کنید، به علت تفاوت در درجه‌ی پارامترها (10^1 و10^5)، مقدار نهایی تابع برازش وابسته به مقدار E بوده و اثرگذاری پارامتری مثل d را بسیار کم کرده است.

در برخی مقالات جهت اصلاح این مشکل از یک ضریب اصلاح استفاده کرده‌اند. که این ضریب جهت برابر کردن یا تعدیل مرتبه استفاده می‌شود.

۳) حتی بعد از اعمال ضرایب اصلاح، اثرگذاری برخی از پارامترها ممکن است در نظر گرفته نشود، به عنوان مثال فرض کنید ضریب اصلاح برای پارامتر E، $w_1 = 10^{-3}$ و ضریب اصلاح برای پارامتر d، $w_2 = 10^1$ باشد. در این صورت $F_1 = 225+200 = 425$ و $F_2 = 425+0 = 425$ خواهد شد یعنی اثرگذاری پارامتر d درنظر گرفته نشده است.

۴) مشکل دیگری که با اعمال ضریب اصلاح پیش می‌آید این است که در هر مسئله متفاوت، ممکن است درجه‌ی مقادیر پارامترها متفاوت باشند. پس در هر بار اجرا یا در هر

مسئله‌ی متفاوت باید ضرایب اصلاح را هم تغییر دهیم.

۵) مسئله‌ی دیگر قابل قیاس نبودن نتایج الگوریتم در دو شبکه‌ی حسگر مجزا است. به عنوان مثال فرض کنید تابع F را در دو شبکه‌ی مجزا، اولی با ۲۰ گره و دومی با ۱۰۰ گره اعمال کردیم. در این صورت اگر برای یکی از حالات مسئله مقدار تابع برازش در شبکه‌ی اول برابر با ۳ × $۱۰^{۵}$ و در شبکه‌ی دوم برابر با ۶ × $۱۰^{۵}$ باشد، به نظر می‌رسد که حالت دوم بهتر است در صورتی که مقایسه در این حالت اشتباه است.

۲,۲ بررسی برخی از توابع برازش استفاده شده در شبکه‌ی حسگر بی‌سیم

در مقاله [4] تابع برازش به صورت زیر تعریف شده است:

$$F = \min\left(\sum_{i=1}^{5} k_i p_i\right) \quad (۱)$$

که در آن k_i وزن (ضریب) و p_i پارامترهای مسئله می‌باشند.

در مقاله‌ی [5] تابع برازش به صورت زیر تعریف شده است:

$$F = w \times (D - distance_i) + (1-w) \times (N-H_i) \quad (۲)$$

که در آن D فاصله‌ی تمامی گره‌ها تا ایستگاه پایه، $distance_i$ فاصله‌ی گره‌ها تا سرخوشه و فاصله‌ی سرخوشه‌ها تا ایستگاه پایه، N تعداد کل گره‌ها، H_i تعداد سرخوشه‌ها و w وزن (ضریب) می‌باشد.

در مقاله‌ی [6] تابع برازش به صورت زیر تعریف شده است:

$$F = w_e.E_s(X) + w_d.D_s(X) + w_l.L_s(X) \quad (۳)$$

که در آن L انرژی هدر رفته، E انرژی مصرفی، D تأخیر و w_e، w_d، w_l ضرایب (وزن‌ها) می‌باشد.

در مقاله‌ی [7] تابع برازش به صورت زیر تعریف شده است:

$$f = \min\{\sum_{\substack{i=1 \\ i\neq 4}}^{7} w_i J_i + w_4 J_4'\} \quad (۴)$$

که در آن w_i به عنوان ضریب و J_i به عنوان پارامترهای تابع برازش می‌باشند.

در مقاله‌ی [8] تابع برازش به صورت زیر تعریف شده است:

$$F = \sum_i (w_i \times f_i) \ , \forall f_i \in \{C, DD, E, SD, T\} \quad (۵)$$

که درآن C مجموع فاصله‌ی هرگره تا سرخوشه‌اش، DD مجموع فاصله‌ی هرنود تا ایستگاه، E انرژی مصرفی، SD پراکندگی سرخوشه‌ها و T تعداد نقل و انتقالات می‌باشد.

همان‌طور که مشاهده می‌کنید، در تمامی توابع برازش فوق از عملگر جمع برای ارتباط پارامترها در تابع استفاده شده است و همچنین بدون توجه به واحد پارامترها، صرفا از یک وزن جهت تعدیل در مرتبه‌ی پارامترها یا یکسان نمودن واحدها استفاده شده است.

۳. اصلاح مشکلات توابع برازش

مشکل اول:

یکسان نبودن واحد و درجه‌ی پارامترهایی که در کنار هم در یک فرمول قرار می‌گیرند.

راه‌حل:

در تمامی الگوریتم‌های تکاملی باید طبق هدفی که برای بهینه‌سازی مسئله داریم یک سری پارامتر از آن استخراج کنیم. این پارامترها از مقادیر خام مسئله ساخته می‌شود. به عنوان مثال مقادیر خام در مسئله‌ی مرتبط با شبکه‌ی حسگر بی‌سیم می‌تواند مختصات مکان هر گره، انرژی اولیه و باقیمانده‌ی هر گره و غیره باشد. سپس بنا بر نیاز و تشخیص سازنده‌ی تابع از مقادیر خام، تحت یک سری روابط ریاضی، پارامترهای

تابع برازش را می‌سازند.

هنگام استفاده از این پارامترها باید به دو نکته توجه داشت:

۱) زمانی که این پارامترها در کنار هم و تحت یک رابطه‌ی ریاضی قرار می‌گیرند باید دقت شود که دارای واحدهای یکسانی باشند. و به دلیل این که یکسان‌سازی تمامی واحدها امری منطقی و امکانپذیر نیست، پس باید آنها را بی‌واحد ساخت. در مقالات بررسی شده در فصل دوم، این بی‌واحدسازی صرفا با ضرب کردن ضریبی انجام می‌گرفت و اندازه‌ی این ضریب‌ها تابعی از اندازه‌ی هر پارامتر در هر مسئله بود.

۲) تفاوت در اندازه و درجه‌ی هر پارامتر است که باعث می‌شود نتوان بطور خام داده‌ها را با یکدیگر ترکیب کرد و نیاز به قرار دادن ضریبی در کنار آنها برای قابل قیاس کردن آنها به وجود می‌آید. ولی با توجه به تفاوت درجات (که مهم‌تر از تفاوت در مقادیر است) در جواب‌های مختلف این ضرایب وابسته و مختص یک مسئله می‌شود.

در این مقاله سعی شده است که دو مشکل فوق (بی‌واحدسازی و تفاوت درجات) با فرآیندی شبیه به نرمال‌سازی رفع شود. طی این فرآیند $\left(\frac{x-\mu}{s^*}\right)$ اعداد بدست آمده که علاوه بر بی‌واحد شدن، درصد زیادی از داده‌ها را در بازه‌ای محدود قرار می‌دهند، که این امر امکان مقایسه‌ی نتایج تابع برازش را حتی بین جواب‌های مختلف در شبکه‌های مختلف ایجاد می‌کند.

یک دلیل شهودی برای نشان دادن لزوم انجام فرآیندی شبیه به نرمال‌سازی:

فرض کنید در دو شبکه‌ی حسگر بی‌سیم مجزا، اعداد اولیه‌ی پارامترها همانند جدول ۱ باشد:

حال این سوال مطرح می‌شود که بین A_1 و B_1 کدام بهتر است؟ اگر پارامترها در هم ضرب شوند آنگاه:

$$A_1 = 9 \times 10^7 \quad , \quad B_1 = 1 \times 10^9$$

با مقایسه‌ی ظاهری به این نتیجه می‌رسیم که B_1 بهتر است. ولی A1 در شبکه‌ی خود از میانگین بالاتر بوده است و اگر تعداد حالات زیاد باشد، حاکی از خوب بودن آن است. ولی برای B1 صادق نیست. زیرا در شبکه‌ی خود، از میانگین کمتر است. پس باید اعداد بعد از اصلاح شدن با یکدیگر مقایسه شوند که ما در اینجا برای مقایسه، پارامترها را از همان ابتدا حول میانگین خودشان دسته‌بندی کردیم تا دیگر نحوه‌ی اصلاح کردن، مستقل از مقادیر اولیه‌ی پارامترها باشد.

جدول ۱: مقادیر فرضی دو پارامتر تابع برازش

	E	K
A_1	3×105	3×102
A_2	2.5×105	4×102
A_3	4×105	2×102
A_4	5×105	3×102
B_1	1×107	1×102
B_2	2.5×107	2×102
B_3	3×107	3×102
B_4	3.1×107	1×102

مشکل دوم:

استفاده از عملگر جمع که وابستگی تأثیر تغییرات یک پارامتر به سایر پارامترها را در نظر نمی‌گیرد.

راه‌حل:

بعد از تعیین پارامترهای مسئله این موضوع پیش می‌آید که چگونه این پارامترها را در قالب یک تابع کلی (تابع برازش) با یکدیگر مرتبط نماییم. به طوری- که اثرگذاری پارامترها بر روی یکدیگر در خروجی تابع نمود پیدا کند. این اصل مهم که طی فرآیند مرتبط‌سازی پارامترها مد نظر است را می‌توان به صورت زیر بیان کرد:

"مشتق جزئی (توان تأثیرگذاری) هر تابع (پارامتر) وابسته به مقادیر سایر توابع (پارامترها) یا خود تابع (پارامتر) است"

مفهوم فوق را می‌توان به این صورت تعبیر کرد که تغییرات $\frac{\partial f_i}{\partial y_i}$ به ازای تغییرات y_jها $(i \neq j)$ صفر نباشد. تعبیر ریاضی عبارت فوق را می‌توان به صورت زیر

نوشت:

$$\frac{\partial}{\partial y_j}(\frac{\partial f}{\partial y_i}) \neq 0 , \qquad i,j \in \{1..n\}, i \neq j \qquad (٦)$$

$$\frac{\partial^2}{\partial y_j \partial y_k}(\frac{\partial f}{\partial y_i}) \neq 0 \quad ,i,j,k \in \{1..n\}, i \neq j \neq k$$

... .(٧)

$$\frac{\partial^{n-1}}{\partial y_{j_1} \dots \partial y_{j_{n-1}}}(\frac{\partial f}{\partial y_i}) \neq 0 \quad ,j_1,..,j_{n-1},i \in \{1..n\} \quad , \quad i \neq j_1 \neq \dots \neq j_{n-1}$$

که یکی از روش‌های برآورده کردن خواسته‌ی فوق ضرب توابع (پارامترها) در یکدیگر است. یعنی $\prod_{i=1}^{n} y_i$ و برای ایجاد تفاوت در توان تأثیرگذاری می‌توان از $y_i^{\propto_i}$ به جای y_i استفاده کرد.

بدیهی است که با جمع توابع (پارامترها) خواص فوق برآورده نمی‌شود. چون اگر

$$F = \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n \qquad (٨)$$

آنگاه

$$\frac{\partial f}{\partial x_1} = \alpha_1 \quad , \quad \frac{\partial}{\partial x_2}(\frac{\partial f}{\partial x_1}) = 0 \quad , \qquad (٩)$$

یعنی تغییرات یک پارامتر، وابسته به بقیه‌ی پارامترها نیست.

بیان این مسئله به زبان بسیار ساده، به این صورت است که فرض کنید تابع برازش دارای پنج پارامتر است که مقادیر آن به صورت زیر می‌باشد:

جدول ۲: مقادیر فرضی پنج پارامتر تابع برازش

	y_1	y_2	y_3	y_4	y_5
حالت اول	۰	۰	۰	۲	۲
حالت دوم	۰/۸	۰/۸	۰/۸	۰/۸	۰/۸

در هر دو حالت فوق $\sum_{i=1}^{5} y_i = 4$ می‌باشد. در صورتی که در حالت اول هیچ ارزشی برای پارامتر اول تا سوم قائل نشده است. یعنی بد بودن مقدار برخی از پارامترها (اگر کم بودن را به بد بودن تعبیر کنیم) هیچ تأثیری بر روی اثرگذاری بقیه‌ی پارامترها در نتیجه‌ی نهایی ندارد.

ولی اگر همین مسئله را توسط $\prod_{i=1}^{5} y_i$ حل نماییم، مشاهده خواهیم کرد که تمامی پارامترها در تعیین مقدار تابع برازش تأثیر خواهند داشت.

۴. شبیه‌سازی

برای درک بهتر مشکلات بیان شده در بخش ۲ و همچنین راه‌کارهای ارائه شده در بخش ۳، مقاله‌ی[8] به صورت تصادفی انتخاب شده و تابع برازش آن پیاده‌سازی شد.

درمقاله‌ی مذکور، تابع برازشی برای خوشه‌بندی شبکه‌ی حسگر بی‌سیم به صورت زیر تعریف شده است:

$$F = \sum_i (w_i \times f_i) \ , \forall f_i \in \{C, DD, E, SD, T\} \ (1) \qquad (١٠)$$

که پارامترهای این تابع به شرح زیر می‌باشد:

۱) DD مجموع فاصله‌ی هر گره تا ایستگاه پایه است که به صورت زیر محاسبه می‌شود:

$$DD = \sum_{i=1}^{m} d_{is} \qquad (١١)$$

و در آن mتعداد گره‌ها در شبکه، d_{is} فاصله‌ی گره‌ی i تا ایستگاه پایه (گره s) می‌باشد.

۲) C مجموع فاصله‌ی هر گره تا سرخوشه‌ی مربوطه و فاصله‌ی سرخوشه‌ها تا ایستگاه پایه می‌باشد که به صورت زیر محاسبه می‌شود:

$$C = \sum_{i=1}^{k} d_{ih} + d_{hs} \qquad (١٢)$$

و در آن d_{ih} فاصله‌ی هر گره‌ی i تا سرخوشه‌ی h و k تعدادگره‌ها در هر خوشه و d_{hs} فاصله‌ی سرخوشه‌ی h تا ایستگاه پایه است.

۳) SD انحراف معیار مختصات سرخوشه‌ها در کل شبکه می‌باشد که به صورت زیر محاسبه می‌شود:

$$\mu = \frac{\sum_{i=1}^{h} d_{cluster_i}}{h} \quad (۱۳)$$

$$SD = \sqrt{\sum_{i=1}^{h} (\mu - d_{cluster_i})^2} \quad (۱۴)$$

۴) E انرژی لازم جهت نقل و انتقال پیغام از خوشه‌ها تا ایستگاه پایه می‌باشد که به صورت زیر محاسبه می‌شود:

$$E = \sum_{j=1}^{k} E_{T_{jh}} + k \times E_R + E_{T_{hs}} \quad (۱۵)$$

و در آن $E_{T_{jh}}$ انرژی لازم جهت انتقال پیغام از k گره‌ی یک خوشه به سرخوشه‌ی مربوطه، $k \times E_R$ انرژی مصرفی برای هرسرخوشه جهت دریافت پیغام از طرف k گره، $E_{T_{hs}}$ انرژی لازم جهت انتقال پیغام از سرخوشه به ایستگاه پایه.

۵) T تعداد نقل و انتقالات می‌باشد.

در نهایت برای ساخت تابع برازش، هر یک از پارامترهای فوق را باید در وزنی ضرب کرده و سپس حاصل‌ضرب‌ها را باهم جمع کنیم، برای ساخت وزن هر پارامتر از فرمول زیر استفاده می‌شود:

$$w_i = w_{i-1} + c_i \Delta f_i \quad (۱۶)$$

$$\Delta f_i = f_i - f_{i-1} \ , \ c_i = \frac{1}{1+e^{f_i}} \quad (۱۷)$$

در این شبیه‌سازی سعی شده است تمامی مراحلی که نویسنده در مقاله‌ی [8] بیان کرده است، رعایت شود. که از جمله‌ی آن می‌توان به استفاده از الگوریتم ژنتیک به عنوان الگوریتم به کار گیرنده‌ی تابع برازش، نحوه‌ی تعریف کروموزوم‌های الگوریتم ژنتیک وغیره نام برد.

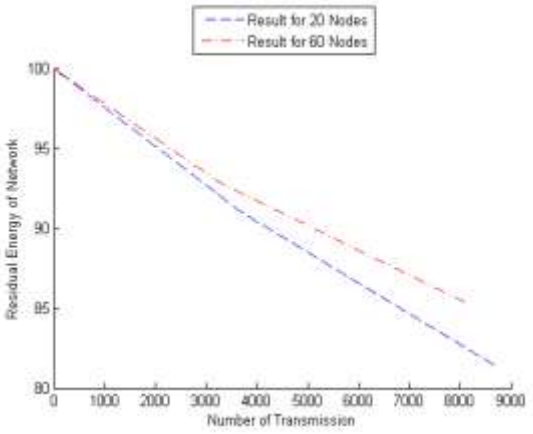
شبکه‌ی حسگر بی‌سیم توسط تابع برازش فوق، خوشه‌بندی شده و به صورت اتفاقی دو کروموزوم از یک نسل انتخاب شده است که در جدول ۳ قابل مشاهده می‌باشد.

مقادیر تابع برازش که براساس جدول ۳ و رابطه‌ی ۱ برای هردو کروموزوم حساب شده است، به صورت زیر می‌باشد:

F1 = 39.04 و F2 = 39.34

جدول ۳: مقادیر حاصل از شبیه‌سازی

پارامترها	E	T	DD	SD	C
پارامترهای کروموزوم۱	۱/۱	۷۳۸	/۳۶۳ ۱۵۶۵	/۸۵۸ ۳۹	/۹۱۸ ۶۵۷
پارامترهای کروموزوم۲	/۰۲۱ ۳	۷۱۱	/۳۶۳ ۱۵۶۵	۰/۵۹۱	/۰۹۸ ۷۴۱
وزن پارامترها	/۰۱۵ ۰	۰/۰۳	۰/۰۰۲	۰/۰۱۵	۰/۰۲



شکل ۱: نمودار مقایسه‌ای انرژی باقیمانده‌ی شبکه بر حسب تعداد تراکنش برای ۲۰ گره و ۶۰ گره

بررسی مشکلات تابع برازش شبیه‌سازی شده براساس نتایج بدست آمده از شبیه‌سازی:

۱) همان‌طورکه مشاهده می‌کنید پارامترهای موجود در این تابع دارای واحدهای متفاوتی می‌باشند که بدون توجه به تفاوت واحدها، درکنار هم وتحت یک تابع قرار گرفته‌اند، به عنوان نمونه، انرژی بر حسب kgm^2/s^2 و DD یا C برحسب m(متر) می‌باشد.

۲) اعداد تولید شده دارای درجات متفاوتی هستند و اثرگذاری برخی از پارامترها نسبت به بقیه نادیده گرفته می‌شود مثل اثرگذاری E=1.1 نسبت به DD=1565.363 یا C=657.918. که در این مقاله برای رفع این مشکل از یک وزن طبق رابطه‌ی ۲ استفاده کرده است.

۳) در طی شبیه‌سازی، یک‌بار تعداد گره‌های شبکه ۲۰ عدد و بار دیگر ۶۰ عدد در نظر گرفته شد، در هر مرحله نمودار انرژی مصرفی بر حسب تعداد نقل و انتقالات را رسم کرده و نتیجه‌ی نهایی در شکل ۱ نمایش داده شد.

۴) همان‌طور که از شکل ۱ مشخص است این دو نمودار قابل قیاس با یکدیگر نیستند. زیرا هرکدام دارای شرایط متفاوت نسبت به یکدیگر می‌باشند.بنابراین همیشه این مشکل وجود خواهد داشت که درحالتی‌که سایز شبکه و بالطبع بقیه‌ی پارامترها بایکدیگر تفاوت دارند هرگز دوشبکه را نمی‌توان از لحاظ کارکرد، بایکدیگر مقایسه کرد. یا به عنوان مثال زمانی که بخواهیم این تابع برازش را با تابع برازشی دیگر مقایسه کنیم راهی به غیر از پیاده‌سازی هر دو تابع در شرایط یکسان نخواهیم داشت و نمی‌توانیم صرفا از نمودارهای موجود در هر مقاله استفاده کنیم.

۵) با توجه به مقادیر F1 و F2 که براساس جدول۳ بدست آمدند، مشاهده می‌شود که ارزش کروموزوم اول کمتر از کروموزوم دوم است درصورتیکه از روی مقادیر پارامترهای جدول۳ کاملا بدیهی می‌باشد که کروموزوم اول در مقایسه با کروموزوم دوم، هم از لحاظ پراکندگی سرخوشه‌ها(SD) نسبت به مقدار C وضعیت بهتری دارد و هم از لحاظ تعداد نقل وانتقال پیام‌ها نسبت به انرژی مصرفی بسیار بهینه‌تر می‌باشد، درصورتیکه تابع برازش فعلی کروموزوم دوم را انتخاب می‌کند.

راه حل :

برای حل مشکلات ۱و۲و۳ کافیست تمامی مقادیر پارامترها با استفاده از فرمول $\left(\frac{x-\mu}{s^*}\right)$ نرمال شوند. زیرا طی فرآیند نرمال شدن، تمامی داده‌ها علاوه بر تقسیم بر انحراف معیار خود که مشکل بی‌واحدسازی را حل می‌کند،با تفاضل از میانگین داده‌ها و تقسیم بر انحراف معیار در بازه‌ای محدود قرار می‌گیرند. که این امر حتی امکان مقایسه‌ی نتایج تابع برازش را بین جواب‌هایمختلف در شبکه‌های مختلف ایجاد می‌کند.

که در این شبیه سازی مقادیر نرمال شده‌ی هر پارامتر به صورت زیر خواهد شد:

$$Z(C)_i = \frac{\bar{C} - C_i}{\sqrt{\frac{1}{k-1}\sum_{j=1}^{k}(C_j - \bar{C})^2}}$$

$$Z(T)_i = \frac{\bar{T} - T_i}{\sqrt{\frac{1}{k-1}\sum_{j=1}^{k}(T_j - \bar{T})^2}}$$

$$Z(E)_i = \frac{\bar{E} - E_i}{\sqrt{\frac{1}{k-1}\sum_{j=1}^{k}(E_j - \bar{E})^2}}$$

و برای حل مشکل ۴ کافیست به جای عملگر جمع برای ارتباط بین پارامترها از عملگر ضرب استفاده شود. در این صورت مقدار هرپارامتر به طور مستقل بر روی تابع برازش اثرخواهد داشت. به عنوان نمونه اگر نتایج شبیه سازی در جدول۳ در یکدیگر ضرب شوند، کروموزوم اول انتخاب خواهد شد چون :

$$F_1 = 8.99 \quad , \quad F_2 = 0.39$$

۵. نتیجه‌گیری

در کلیه‌ی الگوریتم‌های تکاملی مهم‌ترین بخش آن تابع برازش می‌باشد چراکه این تابع مسئولیت اصلی ارزیابی بهینگی جواب‌های مسئله را بعهده دارد، این الگوریتم‌ها در زمینه‌های مختلفیاز جمله شبکه‌ی حسگر بی‌سیم مورد استفاده قرار می‌گیرد.

دراین مقاله مشکلاتی که اکثر مقالات هنگام ساخت و کنارهم قرار دادن پارامترها داشتند را بررسی کرده و راه حل‌هایی برای رفع آنها ارائه شد.

این مشکلات اولا شامل یکسان نبودن واحد و درجه‌ی پارامترهای یک رابطه و ثانیا عدم تاثیرگذاری مقادیر پارامترها در نتیجه‌ی نهایی تابع برازش می‌باشد که برای حل این مشکلات فرآیند نرمال‌سازی داده‌ها و ضرب پارامترها در یکدیگر پیشنهاد شده است.

مراجع

[1] Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. IEEE Communications Magazine, 8(40):102–114, August 2002.

[2] Akyildiz, TommasoMelodia, and K. Chowdhury. A survey on wireless multimedia sensor networks. Computer Networks, 51:921–960, 2007.

[3] C. Laurent, Didier Helal, Lucille Verbaere, Armin Wellig, and JulienZory. Wireless sensor networks devices: Overview, issues, state of the art and promising technologies. ST Journal of Research, 4(1):8–11, June 2007.

[4] A. P. Bhondekar, R.Vig, M. LalSingla, C. Ghanshyam, P. Kapur, "Genetic Algorithm Based Node Placement Methodology For Wireless Sensor Networks", Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong.

[5] S. Jin, M. Zhou, and A. S. Wu, "Sensor network optimization using a genetic algorithm," in Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics, 2003.

[6] J. Fan, D.J.Parish,"Using a Genetic Algorithm to optimize the Performance of a Wireless Sensor Network ", ISBN: 1-9025-6016-7 © ,2007 PGNet

[7] K. P. Ferentinos, T. A. Tsiligiridis, "Adaptive design optimization of wireless sensor networks using genetic algorithms", Computer Networks 51 (2007) 1031-1051

[8] S.Hussain, A. W.Matin, O. Islam, "Genetic Algorithm for Hierarchical Wireless Sensor Networks", JOURNAL OF NETWORKS, VOL. 2, NO. 5, SEPTEMBER 2007