

8 Битийн Микроконтроллэрт Зориулсан Дифференциал Хөтлөгчит Хөдөлгөөнт Роботын Өөрийн Байршлыг Тодорхойлох Алгоритм

Ө.Батбаяр

Электроник Холбооны Инженерчлэлийн Тэнхим
Монгол Улсын Их Сургууль,
Хэрэглээний Шинжлэх Ухаан Инженерчлэлийн
Сургууль
batbayar.unursaikhan@gmail.com

Хураангуй – Энэхүү ажил нь дифференциал хөтлөгчит хөдөлгөөнт роботын ажиллагааг илүү ухаалаг болгоход зориулж, түүний өөрийн байршлыг олох алгоритмыг хялбаршуулахад чиглэгдсэн болно. Хөдөлгөөнт роботын өөрийн байршил тодорхойлох аргуудаас тасалгааны нөхцөлд хэрэглэж болох найдвартай бөгөөд бага өртөгтэй арга нь эргэлтын кодер ашиглах арга юм. Эргэлтын кодероор өөрийн байршлыг тодорхойлох алгоритмууд нь хөвөгч цэгтэй бутархай тооны үржих, хуваах болон синус, косинус үйлдэл ашигласан байдаг. Энэ нь өндөр хүчин чадал бүхий процессор хэрэглэх шаардлага бий болгодог. Бидний энэ ажлаараа бага хүчин чадалтай процессорт зориулж хөвөгч цэгтэй бутархай тооны үйлдэл ашиглахгүй хөдөлгөөнт роботын өөрийн байршлыг тооцоолох алгоритмыг дэвшүүлж байна. Алгоритм нь байршлыг тооцоолохдоо зөвхөн бүхэл тооны нэмэх болон хасах үйлдэл ашиглах юм. Уг алгоритм нь хол зам явах тусам алдаа нэмэгдэж байсан. Алгоритмыг Матлаб симуляцийн програм ашиглан туршсан.

Түлхүүр үг – хөдөлгөөнт робот, байршил тогтоох, координат олох, роботик

I. ОРШИЛ

Энэхүү ажлын зорилго нь дифференциал хөтлөгчит хөдөлгөөнт роботын өөрийн байршлыг тодорхойлох юм. Үүнийг хийхэд тавигдах шаадлага нь хурд багатай микроконтроллёр, нарийвчлал сайтай координат олох явдал юм. Өмнө бид хөдөлгөөнт роботыг явах замынх нь хэмжээг өгч удирддаг байсан. Ингэж удирдахад замын халтиргаа, хотгор, гүдгэрээс хамааран робот хажуу тийш шилжилт хийн өгсөн даалгавраас өөр замаар явдаг бөгөөд энэхүү үүссэн алдааг мэдэх боломжгүй байдаг. Иймээс хөдөлгөөнт робот очих ёстой газраа очиж чаддаггүй. Үүнийг шийдэх арга нь хөдөлгөөнт роботын өөрийн байршлыг тодорхойлох юм. Ингэж тодорхойлсноор хөдөлгөөнт робот явалт хийх үедээ хэр зэрэг алдаа гаргасан гэдгээ мэдэх боломжтой болж улмаар алдаагаа засах алгоритмуудыг хөгжүүлэх боломжтой юм. Бид өмнө математик тооцоолол бүхий байршил тодорхойлох ГАЗ алгоритм

Ө.Батбаяр О.Золжаргал

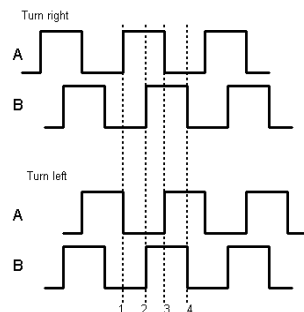
Электроник Холбооны Инженерчлэлийн Тэнхим
Монгол Улсын Их Сургууль,
Хэрэглээний Шинжлэх Ухаан Инженерчлэлийн
Сургууль

туршиж үзэж байсан боловч микроконтроллэрын хурд хүрэлцэхгүй улмаас хөдөлгөөнт робот хурдтай явах үед алдаатай бодож байсан. Одоогоор дээрх математик тооцоолол бүхий ГАЗ алгоритмыг хурдтай ARM процессор бүхий хавтан дээр туршилтийг үргэлжлүүлэн гүйцэтгэж байна. Тиймээс хурд багатай микроконтроллэрт санах ойгоос өгөгдөл уншин зөвхөн нэмэх үйлдэл гүйцэтгэн хөдөлгөөнт роботын байршил тодорхойлох дараах алгоритмыг боловсруулсан.

Улаанбаатар
3 сарын 18, 2013

II. ОНОЛ

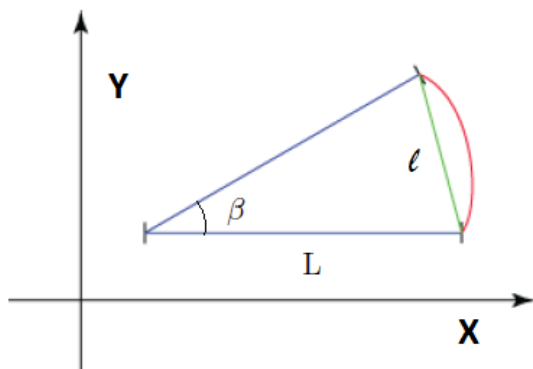
Эргэлтийн кодер – Энэ нь өгөгдөлөө импульс хэлбэрээр контроллэрт дамжуулна. Эдгээр дамжуулсан импульсын тоогоор энкодерын туулсан замын уртыг тодорхойлно. Бидний ашиглаж буй энкодер нь нэг бүтэн эргэхэд 500 импульс дамжуулах ба энкодерын дугуй нь 20 см - ийн диаметртэй. Эндээс бидний нэг алхамын урт болох нэг импульсын явах зам нь 0,04 см болно. Зураг 1 - д энкодерын өгөгдөлөө хэрхэн дамжуулж буйг харуулсан байна.



Зураг 1 эргэлтийн кодерын өгөгдөл дамжуулалт

Энкодер нь дээрх зурган дээр харуулсан шиг хоёр импульсын цуваа дамжуулах бөгөөд эхний импульсын тасалдал орж ирэхэд дараагийн импульс өндөр эсвэл нам байгаагаас харгалзан энкодерын дугуйны урагш эсвэл хойш эргэж байгааг тодорхойлно.

Загварчлал –Дифференциал хөтлөгчит хөдөлгөөнт роботын хөдөлгөөнийг загварчлахдаа хөдөлгөөнт роботын алхамыг дараах байдалтай загварчилна.



Зураг 2. Эргэлтийн кодериин загварчлал

Хөдөлгөөнт роботын нэг алхам маш бага урттай нум үүсэх бөгөөд үүнийг шулуун маягаар загварчилна. Ингэснээр бидний тооцоолол хялбар болно. Зураг 2 –т үзүүлсэн зурган дээрх β өнцөг нь роботын баруун дугуй нэг импульс буюу $\ell = 0,04$ см туулахад үүсэх өнцөг юм. Бидний ашиглаж буй энкодерын нарийвчлал тийм их биш учир бидэнд маш их нарийвчлалтай бодох шаардлага байхгүй. Хөдөлгөөнт роботын хоёр хөтлөгч зөрүүтэй хурдтайгаар урагшилбал хурдны зөрүүнээс робот хазайна. Иймээс роботын хазайлтыг хоёр энкодерийн зөрүүнээс бодно. Роботын нэг алхам бүр дээр координат олох үйлдлийг гүйцэтгэнэ.

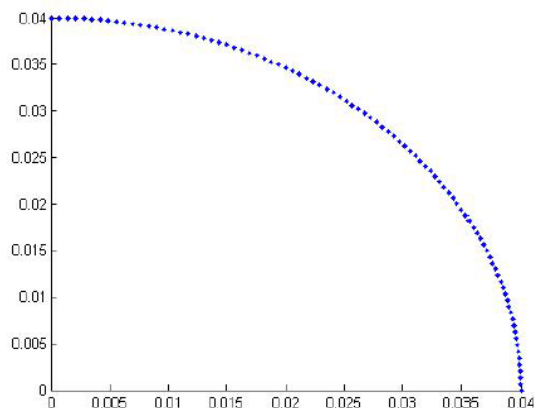
Алгоритм –Хөдөлгөөнт робот координатын хавтгай дээр ямар өнцөг үүсгэн байрлаж буйгаас хамааран дараагийн алхамыг хийхэд нэг алхамын урт нь X , Y координат дээр ямар хэмжээгээр нэмэгдэхийг Зураг 3 - аас харж болно.

Үүнийг тооцоолоход роботын нэг алхам хийсний дараах өнцгөөр нэг алхамын уртын X , Y тэнхлэг дээрх проекцыг бодон X , Y координат дээр нэмж өгнө. X , Y тэнхлэгийн өнцөгт харгалзан нэмэгдэх утгууд үүнд 0 –ээс 90 градусын хооронд байрлах үед нэг алхам хийхэд үүсэх роботын өнцгөөс хамаарсан X , Y – ийн проекцын утгуудыг агуулсан хоёр хүснэгт хадгална.

$$X(n(\gamma)) = L * \sin(\gamma) \quad (1)$$

$$Y(n(\gamma)) = L * \cos(\gamma) \quad (2)$$

γ - хөдөлгөөнт роботын нэгж алхамын хийхэд өөрчлөгдөх өнцөг L - Хоёр хөтлөгчийн хоорондох зай n - Хүснэгтийн дугаар Бусад өнцөгт харгалзах утгууд нь тэмдгээрээ ялгаатайгаар давхцах юм.



Зураг 3. Нэг алхамд харгалзах утга

Үүнийг роботын нэг дугуйны хувьд бодон тухайн дугуйны координатыг олно. Улмаар нөгөө дугуйны координатыг олж болно. Зураг 3 - д харуулсан зурагт 90 ялгаатай утга байгаа учир бутархайгаар илэрхийлэгдэх өнцгийг тоймлон роботын өнцгийг бүхэл тоогоор олно. Роботын координатыг дараах томъёогоор тодорхойлно

$$0 \leq \alpha < 90, X = X + x(\alpha), Y = Y + y(\alpha) \quad (3)$$

$$90 \leq \alpha < 180, X = X - x(\alpha), Y = Y + y(\alpha) \quad (4)$$

$$180 \leq \alpha < 270, X = X - x(\alpha), Y = Y - y(\alpha) \quad (5)$$

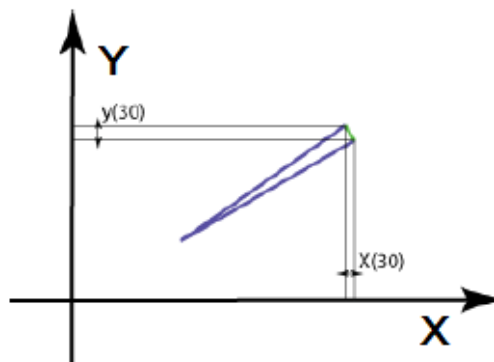
$$270 \leq \alpha < 360, X = X + x(\alpha), Y = Y - y(\alpha) \quad (6)$$

α - Роботын өнцөг

X - Баруун дугуйны x координат

Y - Баруун дугуйны y координат

x , y - Роботын өнцөгт харгалзан X , Y координат дээр нэмэгдэх утгуудыг хадгалсан матриц Жишээ нь робот нэг алхамны дараа 30,2 градусын хазайлттай байвал өнцгийг тоймлон дээрх хоёр хүснэгтийн X хүснэгтийн 30 дахь элементийг X дээр нь нэмж өгнө харин Y хүснэгтийн 30 дахь элементийг Y дээр нь нэмж өгнө. Үүнийг Зураг 4 - д харууллаа.



Зураг 4. Роботын шилжилтын нэг алхам

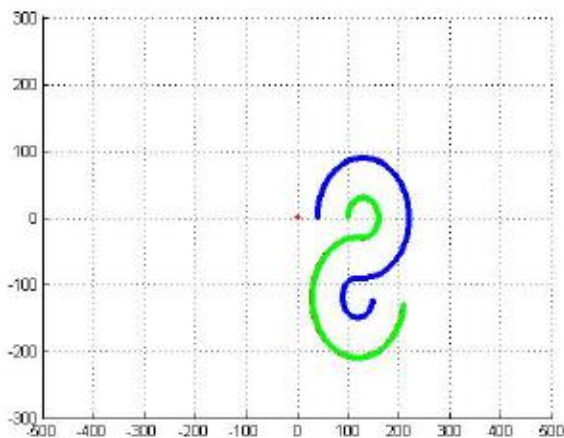
Роботын өнцөг 30 градус байхад үед нэг алхахад X, Y дээр нэмэгдэх утга Эдгээр үйлдлийг хийхэд роботын өнцөг мэдэгдсэн байх шаардлагатай юм. Роботын өнцгийг тодорхойлохдоо хоёр дугуйны энкодерын зөрүүнээс олно. Нэг импульс явахад үүсэх өнцөг нь 360 градусыг робот нэг бүтэн эргэхэд энкодерийн тоолох импульсын тоонд харьцуулж олно.

$$\beta = \frac{360}{n} \quad (7)$$

n - Робот нэг бүтэн эргэхэд энкодерийн тоолох импульсын тоо

III. ТУРШИЛТ

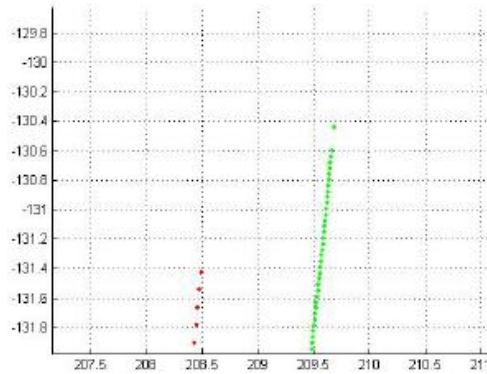
Энэхүү ажлын туршилтыг хийхдээ хөдөлгөөнт роботыг MATLAB дээр загварчлан шалгасан. Ингэж шалгахдаа үнэнтэй мэлээн дөхүү гарах хурдтай процессор дээр ашиглаж болох ГАЗ алгоритмтай харьцуулан шалгасан. Доорх зургуудад ногооноор бидний зохиосон алгоритмаар бодуулж гаргасан зам. Харин улаан болон цэнхэрээр ГАЗ алгоритмаар гаргасан зам. Эдгээрийг харьцуулж хэд хэдэн зам явуулж үзэхэд дараах дүгнэлтүүд хийгдсэн. 8 – ийн тоог зурж үзсэн үр дүнг Зураг -5 болон Зураг -6 - д харуулсан байна.



Зураг 5. 8-ийн тоо зам

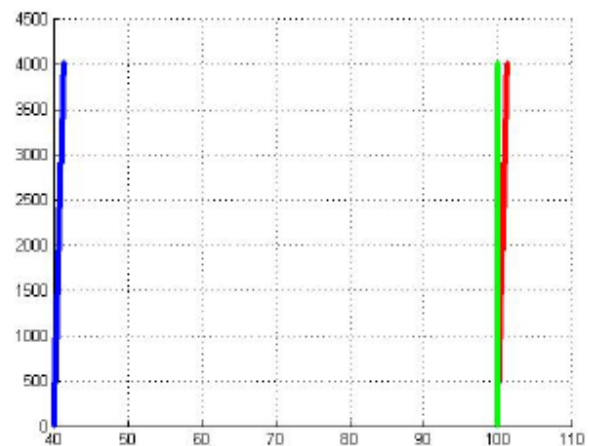
Энэ нь эргэлт хийн 7100 импульс буюу 284 см зам туулахад гарах алдаа нь X –ийн 1 см, Y –ийн 1 см гарч байна. Шулуун зам явуулж туршиж үзсэн үр дүнг Зураг -7 болон Зураг 8 -д харуулсан байна. Шулуун зам явахад манай алгоритм маш бага алдаатай явж байна. Энэ нь өнцгийг тоймлон авснаас шалтгаалан үүсч байна. Энэ нь 100000 импульс буюу 4000 см зам туулахад гарах тооцооллын алдаа нь X –ийн 1 см, Y –ийн хувьд алдаагүй байна. Муруй замаар явуулж

туршсан туршилтийн үр дүнг Зураг 9 болон Зураг 10 - т харуулсан байна.

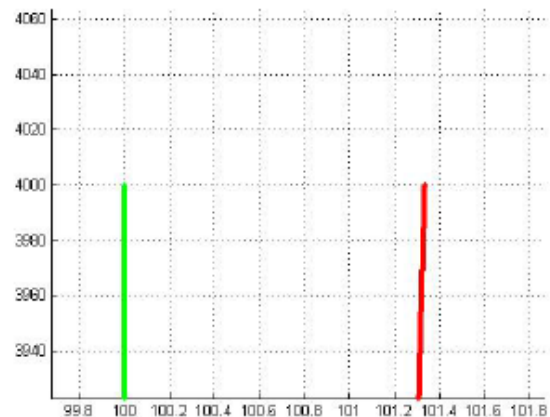


Зураг 6. 8-ийн тоо замыг томруулсан

Роботыг муруй замаар 7000 импульс буюу 280 см зам явуулж шалгасан. Дээрх зургаас ажиглахад гарах алдаа нь X - ийн 2.5 см, Y - ийн 1 см алдаа гарч байна.

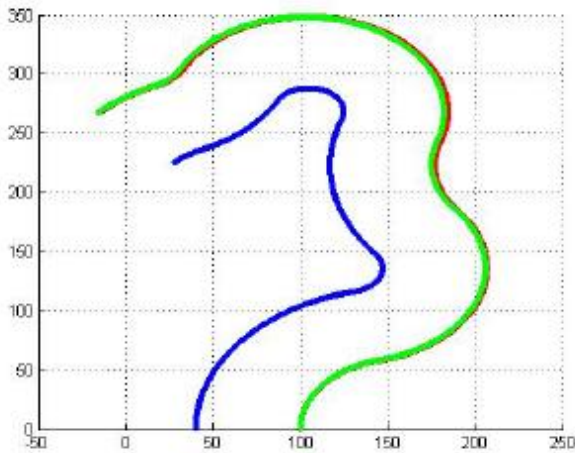


Зураг 7. Шулуун зам

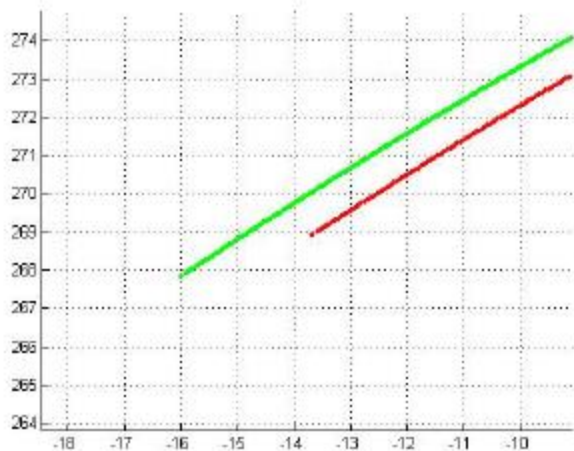


Зураг 8. Шулуун замыг томруулсан

Бид координат хадгалах хувьсагчийг тэмдэггүй 32 битээр хадгална. Ингэхэд 4'294'967'295 доторх тоог илэрхийлж чадна. Бид бодолтоо таслалын ард зургаан оронгын нарийвчлалтай авахад 42 метр зам бодох чадвартай болох юм. Математик үйлдэл бүхий алгоритмтай харьцуулбал. Өнцөг бодох функц ижил учир үүнийг харьцуулсангүй. Энэ 1248 цикл буюу 160 мкс байна. Энэ нь бид хамгийн ихдээ 6,25 эргэлт/с хурдтай эргэхэд бидний өнцөг бодох бодолт үнэн зөв буюу микроконтроллер бодож амжина. Дэвшүүлж буй алгоритмын боловсруулах хугацаа нь 102 цикл буюу 13,3 мкс. Энэ нь хамгийн ихдээ 30 м/с хурдтай хөдөлгөөнт роботын явалтанд бодож амжих юм.



Зураг 9. Муруй зам



Зураг 10. Муруй замыг томруулсан

	төрөл	урт(см)	алдаа(см)
1	8-ын тоо	284	X=1, Y=1
2	Шулуун	4000	X=1, Y=0
3	Муруй	280	X=2.5, Y=1

Хүснэгт 1. Алдааны харьцуулалт

IV. ДҮГНЭЛТ

Энэ алгоритм нь их хялбар бөгөөд бага хүчин чадалтай контроллер ашиглан хөдөлгөөнт роботын байршилийг олж чадсан. Энэ алгоритм нь хөдөлгөөнт роботын чиглэсэн өнцөг зөв байхыг шаардах бөгөөд өнцөг бодоход хөвөгч цэгтэй тоон үйлдэл ашиглана. Гэвч өнцөг хөдөлгөөнт роботын координаттай харьцангуй удаанаар өөрчлөгдөх хэмжигдэхүүн учир хөвөгч цэгтэй тоог бодоход асуудал гарахгүй. Энэхүү туршилтаг матлаб програм дээр симуляц хийн шалгаж алдааг үнэлсэн. Байршил тогтоох хугацаа нь хөвөгч цэгтэй бутархай тооны үйлдэл ашиглан тооцоолох агаас 10 дахи хурдтай байсан. Үүнээс үүдэн бага хүчин чадал бүхий микроконтроллэрт хэрэгжүүлэх боломжтой гэж үзэж байна.

ТАЛАРХАЛ

Энэхүү ажлын туршилтийг хийхэд туслалцаа үзүүлсэн ABU Roboson 2013 тэмцээнд оролцож буй Юнител багийн хамт олонд баярлалаа. 2013 оны багийн хамт олны туслалцаатайгаар энэхүү алгоритмыг туршиж ажиллуулж үзлээ.

АШИГЛАСАН НОМ

- [1] Ganbat G, Nanzadragchaа D, Bayarpurev M. "Differential wheeled robot self localization algorithm based on timing information", MMT 2013, pages 103 – 108, 2013.
- [2] Ojeda L. and Borenstein J. "Methods for the reduction of odometer errors in over-constrained mobile robots", Autonomous Robots, Vol.16 No.3, pages 273-286, 2004.
- [3] Edouard Ivanjko, Ivan Petrovic and Nedjeljko Peric. "An approach to odometer calibration of differential drive mobile robots", In International Conference on Electrical Drives and Power Electronics, pages 519–523, 2003.

№	Замын	Замын	Гарсан
---	-------	-------	--------