IMUFactorExample1 code review

연세대학교 CILAB 김태영

Contents

- GTSAM이란?
- Factor graph란?
- IMUFactorExample1 알고리즘 흐름도
- IMUFactorExample1 code review (Line by line)

GTSAM이란?

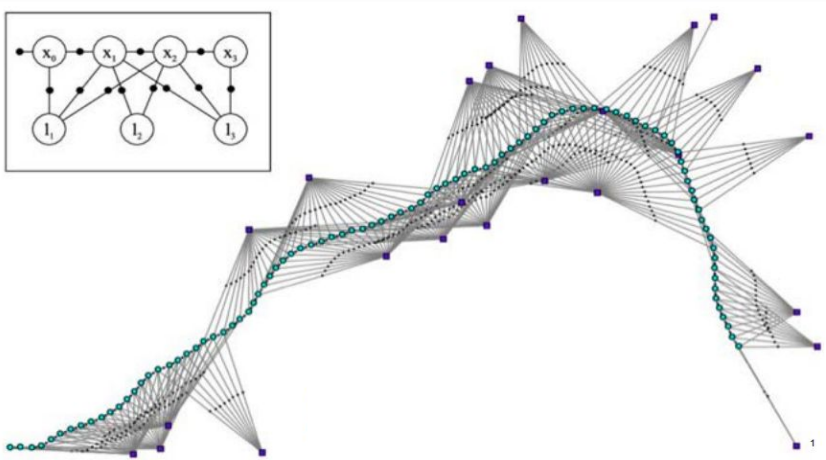
- Georgia Tech에서 만든 센서퓨전 최적화 문제를 풀기 위해 만든 오픈소스 라이브러리
- 최적화를 Factor graph로 모델링을 진행해서 문제를 해결
- IMU preintegration이 모듈로 만들어져 있음.
- 여러 Robotics 환경에서 활용 : SLAM, Robot localization, SFM(Structure from Motion), LQR control, Forward / Inverse kinematics..

https://gtsam.org/

Factor graph란?

2가지 종류의 Node로 구성된 Probabilistic graphical model

- State 또는 Value : 구하고자 하는 parameter
- Factor : Measurement 또는 prior를 통해 얻어진 constraints



$$\mathcal{X}_k^\star \doteq \arg\min_{\mathcal{X}_k} -\log_e p(\mathcal{X}_k|\mathcal{Z}_k)$$

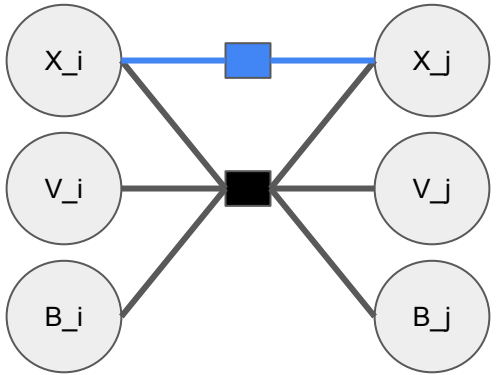
$$= \arg\min_{\mathcal{X}_k} \|\mathbf{r}_0\|^2_{\mathbf{\Sigma}_0} + \sum_{(i,j)\in\mathcal{K}_k} \|\mathbf{r}_{\mathcal{I}_{ij}}\|^2_{\mathbf{\Sigma}_{ij}}$$

https://youtu.be/b7DG4N23o8g

IMUFactorExample code 알고리즘 흐름도

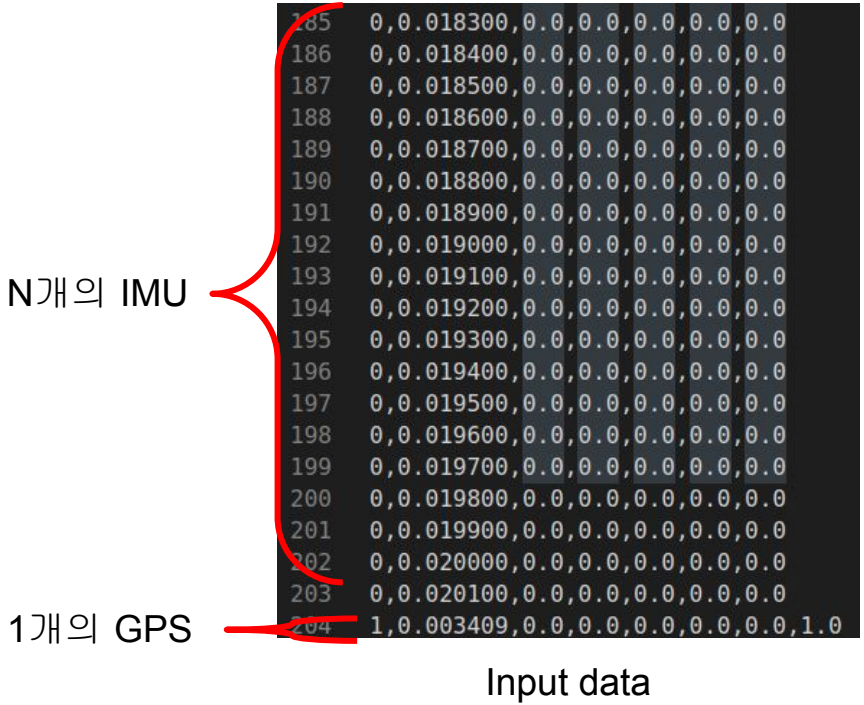
목표 : GPS + IMU fusion을 통한 IMU state 예측 (카메라는 사용하지 않음)

- State : IMU state (IMU pose, IMU velocity, IMU bias)
- Factor : IMU measurement (각속도, 가속도) >> preintegration IMU factor
  GPS measurement (position) >> GPS factor



IMUFactorExample code 알고리즘 흐름도

1. 변수 초기화
2. Input data 불러오기
3. N개의 IMU에 대해 IMU measurement integration
4. Factor 추가 (IMU preintegration factor, GPS factor)
5. ISAM을 활용한 최적화 수행
6. 최적의 state와 GT의 정량적 비교
7. 3~6 과정 반복



Input data

IMUFactorExample code review

- 초기화 과정

```
121  int main(int argc, char* argv[]) {
122    string data_filename, output_filename;
123
124    bool use_isam = false;
125
126    po::variables_map var_map = parseOptions(argc, argv);
127
128    data_filename = findExampleDataFile(var_map["data_csv_path"].as<string>());
129    output_filename = var_map["output_filename"].as<string>();
130    use_isam = var_map["use_isam"].as<bool>();
131
132    // parseOptions에서 정의한 ISAM2 사용 여부에 따라 결정
133    ISAM2* isam2 = 0;
134    if (use_isam) {
135      printf("Using ISAM2\n");
136      ISAM2Params parameters;
137      parameters.relinearizeThreshold = 0.01;
138      parameters.relinearizeSkip = 1;
139      isam2 = new ISAM2(parameters);
140
141    } else {
142      printf("Using Levenberg Marquardt Optimizer\n");
143    }
144
```

필요한 parameter 정의

- Input data 경로
- 출력 파일 이름
- ISAM 사용 여부

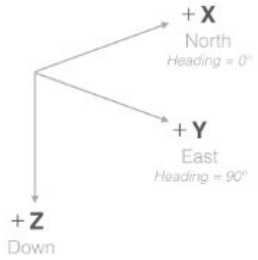
ISAM2 초기화

IMUFactorExample code review

- 초기화 과정

```
145   // Set up output file for plotting errors
146   FILE* fp_out = fopen(output_filename.c_str(), "w+");
147   fprintf(fp_out,
148           "#time(s),x(m),y(m),z(m),qx,qy,qz,qw,gt_x(m),gt_y(m),gt_z(m),gt_qx,"
149           "gt_qy,gt_qz,gt_qw\n");
150
151   // Begin parsing the CSV file.  Input the first line for initialization.
152   // From there, we'll iterate through the file and we'll preintegrate the IMU
153   // or add in the GPS given the input.
154   ifstream file(data_filename.c_str());
155   string value;
156
157   // Format is (N,E,D,qX,qY,qZ,qW,velN,velE,velD)
158   // NED = North East Down 좌표계이기 때문에 그냥 x,y,z로 생각해도 무방할듯
159   Vector10 initial_state;
160   getline(file, value, ',');  // i
161   for (int i = 0; i < 9; i++) {
162     getline(file, value, ',');
163     initial_state(i) = stof(value.c_str());
164   }
165   getline(file, value, '\n');
166   initial_state(9) = stof(value.c_str());
167   cout << "initial state:\n" << initial_state.transpose() << "\n\n";
```

출력 파일 초기화



NED frame

State (최적화 대상) 초기화

IMUFactorExample code review

- Factor graph에 Prior 정보 추가

```
169   // Assemble initial quaternion through GTSAM constructor
170   // ::quaternion(w,x,y,z);
171   Rot3 prior_rotation = Rot3::Quaternion(initial_state(6), initial_state(3),
172                                          initial_state(4), initial_state(5));
173   Point3 prior_point(initial_state.head<3>());
174   Pose3 prior_pose(prior_rotation, prior_point);
175   Vector3 prior_velocity(initial_state.tail<3>());
176   imuBias::ConstantBias prior_imu_bias;  // assume zero initial bias
177
178   // IMU의 초기 value를 설정
179   Values initial_values;
180   int correction_count = 0;
181   initial_values.insert(X(correction_count), prior_pose);
182   initial_values.insert(V(correction_count), prior_velocity);
183   initial_values.insert(B(correction_count), prior_imu_bias);
184
185   // Assemble prior noise model and add it the graph.`
186   auto pose_noise_model = noiseModel::Diagonal::Sigmas(
187       (Vector(6) << 0.01, 0.01, 0.01, 0.5, 0.5, 0.5)
188           .finished());  // rad,rad,rad,m, m, m
189   auto velocity_noise_model = noiseModel::Isotropic::Sigma(3, 0.1);  // m/s
190   auto bias_noise_model = noiseModel::Isotropic::Sigma(6, 1e-3);
191
192   // Add all prior factors (pose, velocity, bias) to the graph.
193   NonlinearFactorGraph* graph = new NonlinearFactorGraph();
194   graph->addPrior(X(correction_count), prior_pose, pose_noise_model); // pose값
195   graph->addPrior(V(correction_count), prior_velocity, velocity_noise_model); // velocity값
196   graph->addPrior(B(correction_count), prior_imu_bias, bias_noise_model); // bias값
197
```

GTSAM state 초기화

논문[1] : A. State 참고

$$\mathbf{x}_i \doteq [\mathbf{R}_i, \mathbf{p}_i, \mathbf{v}_i, \mathbf{b}_i].$$

```
using symbol_shorthand::B;  // Bias  (ax,ay,az,gx,gy,gz)
using symbol_shorthand::V;  // Vel   (xdot,ydot,zdot)
using symbol_shorthand::X;  // Pose3 (x,y,z,r,p,y)
```

[1] Forster, Christian, et al. "On-manifold preintegration for real-time visual--inertial odometry." *IEEE Transactions on Robotics* 33.1 (2016): 1-21.

IMUFactorExample code review

- Factor graph에 Prior 정보 추가

```
169   // Assemble initial quaternion through GTSAM constructor
170   // ::quaternion(w,x,y,z);
171   Rot3 prior_rotation = Rot3::Quaternion(initial_state(6), initial_state(3),
172                                          initial_state(4), initial_state(5));
173   Point3 prior_point(initial_state.head<3>());
174   Pose3 prior_pose(prior_rotation, prior_point);
175   Vector3 prior_velocity(initial_state.tail<3>());
176   imuBias::ConstantBias prior_imu_bias;  // assume zero initial bias
177
178   // IMU의 초기 value를 설정
179   Values initial_values;
180   int correction_count = 0;
181   initial_values.insert(X(correction_count), prior_pose);
182   initial_values.insert(V(correction_count), prior_velocity);
183   initial_values.insert(B(correction_count), prior_imu_bias);
184
185   // Assemble prior noise model and add it the graph.`
186   auto pose_noise_model = noiseModel::Diagonal::Sigmas(
187       (Vector(6) << 0.01, 0.01, 0.01, 0.5, 0.5, 0.5)
188           .finished());  // rad,rad,rad,m, m, m
189   auto velocity_noise_model = noiseModel::Isotropic::Sigma(3, 0.1);  // m/s
190   auto bias_noise_model = noiseModel::Isotropic::Sigma(6, 1e-3);
191
192   // Add all prior factors (pose, velocity, bias) to the graph.
193   NonlinearFactorGraph* graph = new NonlinearFactorGraph();
194   graph->addPrior(X(correction_count), prior_pose, pose_noise_model); // pose값
195   graph->addPrior(V(correction_count), prior_velocity, velocity_noise_model); // velocity값
196   graph->addPrior(B(correction_count), prior_imu_bias, bias_noise_model); // bias값
197
```

State Noise Model 정의

B. Noise Propagation

E. Bias Model

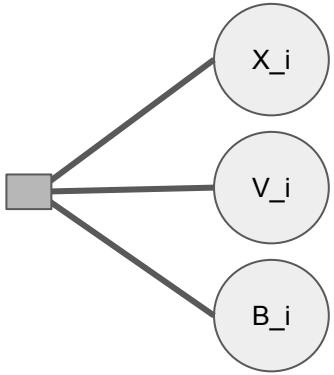
$$\boldsymbol{\eta}_{ij}^{\Delta} \doteq \left[\delta\boldsymbol{\phi}_{ij}^{\mathsf{T}}, \delta\mathbf{v}_{ij}^{\mathsf{T}}, \delta\mathbf{p}_{ij}^{\mathsf{T}}\right]^{\mathsf{T}} \sim \mathcal{N}(\mathbf{0}_{9\times 1}, \boldsymbol{\Sigma}_{ij}).$$

$$\dot{\mathbf{b}}^{g}(t) = \boldsymbol{\eta}^{\mathrm{bg}} \qquad \dot{\mathbf{b}}^{a}(t) = \boldsymbol{\eta}^{\mathrm{ba}}$$

IMUFactorExample code review

- Factor graph에 Prior 정보 추가

```
169  // Assemble initial quaternion through GTSAM constructor
170  // ::quaternion(w,x,y,z);
171  Rot3 prior_rotation = Rot3::Quaternion(initial_state(6), initial_state(3),
172                                         initial_state(4), initial_state(5));
173  Point3 prior_point(initial_state.head<3>());
174  Pose3 prior_pose(prior_rotation, prior_point);
175  Vector3 prior_velocity(initial_state.tail<3>());
176  imuBias::ConstantBias prior_imu_bias;  // assume zero initial bias
177
178  // IMU의 초기 value를 설정
179  Values initial_values;
180  int correction_count = 0;
181  initial_values.insert(X(correction_count), prior_pose);
182  initial_values.insert(V(correction_count), prior_velocity);
183  initial_values.insert(B(correction_count), prior_imu_bias);
184
185  // Assemble prior noise model and add it the graph.`
186  auto pose_noise_model = noiseModel::Diagonal::Sigmas(
187      (Vector(6) << 0.01, 0.01, 0.01, 0.5, 0.5, 0.5)
188          .finished());  // rad,rad,rad,m, m, m
189  auto velocity_noise_model = noiseModel::Isotropic::Sigma(3, 0.1);  // m/s
190  auto bias_noise_model = noiseModel::Isotropic::Sigma(6, 1e-3);
191
192  // Add all prior factors (pose, velocity, bias) to the graph.
193  NonlinearFactorGraph* graph = new NonlinearFactorGraph();
194  graph->addPrior(X(correction_count), prior_pose, pose_noise_model); // pose값
195  graph->addPrior(V(correction_count), prior_velocity, velocity_noise_model); // velocity값
196  graph->addPrior(B(correction_count), prior_imu_bias, bias_noise_model); // bias값
197
```



Equation (26)

$$\mathcal{X}_k^\star \doteq \arg\min_{\mathcal{X}_k} -\log_e p(\mathcal{X}_k|\mathcal{Z}_k)$$

$$= \arg\min_{\mathcal{X}_k} \|\mathbf{r}_0\|^2_{\Sigma_0} + \sum_{(i,j)\in\mathcal{K}_k} \|\mathbf{r}_{\mathcal{I}_{ij}}\|^2_{\Sigma_{ij}}$$

Prior factor를 Factor graph에 추가

IMUFactorExample code review

- IMU preintegration 초기화

```
198  auto p = imuParams();
199
200  std::shared_ptr<PreintegrationType> preintegrated =
201      std::make_shared<PreintegratedImuMeasurements>(p, prior_imu_bias);
202
203  assert(preintegrated);
```

IMUFactorExample cod

- IMU preintegration 추

```
198  auto p = imuParams();
199
200  std::shared_ptr<Preinteg
201      std::make_shared<Preinteg
202
203  assert(preintegrated);
```

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

IMUFactorExample code review

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

IMU sensor modeling 관련 parameter 대입

$$ {}_B\tilde{\boldsymbol{\omega}}_{WB}(t) = {}_B\boldsymbol{\omega}_{WB}(t) + \mathbf{b}^g(t) + \boldsymbol{\eta}^g(t) \tag{27} $$

$$ {}_B\tilde{\mathbf{a}}(t) = \mathbf{R}_{WB}^{\mathsf{T}}(t)\,({}_W\mathbf{a}(t) - {}_W\mathbf{g}) + \mathbf{b}^a(t) + \boldsymbol{\eta}^a(t). \tag{28} $$

$$ \dot{\mathbf{b}}^g(t) = \boldsymbol{\eta}^{\mathrm{bg}} \qquad \dot{\mathbf{b}}^a(t) = \boldsymbol{\eta}^{\mathrm{ba}} $$

B. Bias Correction via First-Order Updates (Appendix)

$$ \hat{\mathbf{b}}_i \leftarrow \bar{\mathbf{b}}_i + \delta\mathbf{b}_i $$

bias_acc_omega_init (6x6)

IMUFactorExample code review

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

```
Matrix3 integrationCovariance;
///< continuous-time "Covariance" describing integration uncertainty
```

Integration Covariance Q_{int} : This is the uncertainty due to modeling errors in the integration from acceleration to velocity and position.

IMUFactorExample code review

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

IMU의 gravity 방향 설정

Example1 code의 data의 경우,

IMU data가 중력을 제거하는 전처리를 거쳤다고 가정

따라서 **중력에 대한 영향을 0으로 설정**

IMUFactorExample code review

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

PreintegratedCombinedMeasurement class에 covariance 관련 값 대입

IMUFactorExample cod

- IMU preintegration 추

```
198  auto p = imuParams();
199
200  std::shared_ptr<Preinteg
201      std::make_shared<Preinteg
202
203  assert(preintegrated);
```

```
87  boost::shared_ptr<PreintegratedCombinedMeasurements::Params> imuParams() {
88    // We use the sensor specs to build the noise model for the IMU factor.
89    double accel_noise_sigma = 0.0003924;
90    double gyro_noise_sigma = 0.000205689024915;
91    double accel_bias_rw_sigma = 0.004905;
92    double gyro_bias_rw_sigma = 0.000001454441043;
93    Matrix33 measured_acc_cov = I_3x3 * pow(accel_noise_sigma, 2);
94    Matrix33 measured_omega_cov = I_3x3 * pow(gyro_noise_sigma, 2);
95    Matrix33 integration_error_cov =
96        I_3x3 * 1e-8;  // error committed in integrating position from velocities
97    Matrix33 bias_acc_cov = I_3x3 * pow(accel_bias_rw_sigma, 2);
98    Matrix33 bias_omega_cov = I_3x3 * pow(gyro_bias_rw_sigma, 2);
99    Matrix66 bias_acc_omega_init =
100       I_6x6 * 1e-5;  // error in the bias used for preintegration
101
102   // grvity vector는 9.81로 대부분 넣는데, example에서 사용하는 data는
103   // 이미 중력이 제거된 상태로 가정하고 있음
104   auto p = PreintegratedCombinedMeasurements::Params::MakeSharedD(0.0);
105
106   // PreintegrationBase params:
107   p->accelerometerCovariance =
108       measured_acc_cov;  // acc white noise in continuous
109   p->integrationCovariance =
110       integration_error_cov;  // integration uncertainty continuous
111   // should be using 2nd order integration
112   // PreintegratedRotation params:
113   p->gyroscopeCovariance =
114       measured_omega_cov;  // gyro white noise in continuous
115   // PreintegrationCombinedMeasurements params:
116   p->biasAccCovariance = bias_acc_cov;      // acc bias in continuous
117   p->biasOmegaCovariance = bias_omega_cov;  // gyro bias in continuous
118   p->biasAccOmegaInt = bias_acc_omega_init;
119
120   return p;
121 }
```

IMUFactorExample code review

- IMU preintegration 초기화

```
198  auto p = imuParams();
199
200  std::shared_ptr<PreintegrationType> preintegrated =
201      std::make_shared<PreintegratedImuMeasurements>(p, prior_imu_bias);
202
203  assert(preintegrated);
```

IMUFactorExample code review

- IMU preintegration 초기화

```
198   auto p = imuParams();
199
200   std::shared_ptr<PreintegrationType> preintegrated =
201       std::make_shared<PreintegratedImuMeasurements>(p, prior_imu_bias);
202
203   assert(preintegrated);
```

컴파일시 선택 가능, default는 Tangent Preintegration이므로,
/cmake/HandleGeneralOptions.cmake에서 옵션을 바꿔야함.

1. Manifold Preintegration: This version keeps track of the incremental NavState ΔX_{ij} with respect to the previous NavState, on the NavState manifold itself. It also keeps track of the $\mathbb{R}^{9\times6}$ Jacobian of ΔX_{ij} w.r.t. the bias. This corresponds to Forster et. al.[1]

2. Tangent Preintegration: This version keeps track of the incremental NavState in the NavState tangent space instead. This is a $\mathbb{R}^9$ vector *preintegrated_*. It also keeps track of the $\mathbb{R}^{9\times6}$ jacobian of the *preintegrated_* w.r.t. the bias.

/cmake/HandleGeneralOptions.cmake

```
option(GTSAM_TANGENT_PREINTEGRATION       "Use new ImuFactor with integration on tangent space" OFF)
```

IMUFactorExample code review

- IMU state 저장 (X,V,B >> NavState, bias)

```
207  // Store previous state for imu integration and latest predicted outcome.
208  // NavState : Pose + velocity
209  NavState prev_state(prior_pose, prior_velocity);
210  NavState prop_state = prev_state;
211  imuBias::ConstantBias prev_bias = prior_imu_bias;
212
213  // Keep track of total error over the entire run as simple performance metric.
214  double current_position_error = 0.0, current_orientation_error = 0.0;
215
216  double output_time = 0.0;
217  double dt = 0.005;  // The real system has noise, but here, results are nearly
218                      // exactly the same, so keeping this for simplicity.
219
```

$$\mathbf{x}_i \doteq [\mathrm{R}_i, \mathbf{p}_i, \mathbf{v}_i, \mathbf{b}_i].$$

```
using symbol_shorthand::B;  // Bias  (ax,ay,az,gx,gy,gz)
using symbol_shorthand::V;  // Vel   (xdot,ydot,zdot)
using symbol_shorthand::X;  // Pose3 (x,y,z,r,p,y)
```

$$\text{NavState } X_b^n \triangleq \{R_b^n, P_b^n, V_b^n\}$$

IMU는 200Hz로 가정

IMUFactorExample code review

- N개의 IMU에 대해 IMU measurement integration

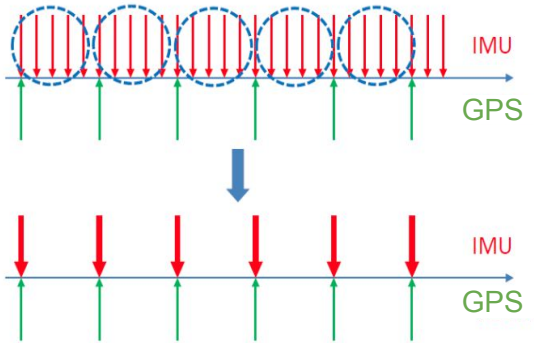
```
220   // All priors have been set up, now iterate through the data file.
221   while (file.good()) {
222     // Parse out first value
223     getline(file, value, ',');
224     int type = stoi(value.c_str());
225
226     if (type == 0) {  // IMU measurement
227       Vector6 imu;
228       for (int i = 0; i < 5; ++i) {
229         getline(file, value, ',');
230         imu(i) = stof(value.c_str());
231       }
232       getline(file, value, '\n');
233       imu(5) = stof(value.c_str());
234
235       // Adding the IMU preintegration.
236       // 가속도, 각속도, dt 값을 넣어서 preintegrated에 계속 쌓는다.
237       preintegrated->integrateMeasurement(imu.head<3>(), imu.tail<3>(), dt);
```

Input data 파일에서
IMU data를 하나씩 꺼내 Vector6에 저장
(gyro, acc)

IMUFactorExample code review

- N개의 IMU에 대해 IMU measurement integration

```
220   // All priors have been set up, now iterate through the data file.
221   while (file.good()) {
222     // Parse out first value
223     getline(file, value, ',');
224     int type = stoi(value.c_str());
225
226     if (type == 0) {  // IMU measurement
227       Vector6 imu;
228       for (int i = 0; i < 5; ++i) {
229         getline(file, value, ',');
230         imu(i) = stof(value.c_str());
231       }
232       getline(file, value, '\n');
233       imu(5) = stof(value.c_str());
234
235       // Adding the IMU preintegration.
236       // 가속도, 각속도, dt 값을 넣어서 preintegrated에 계속 쌓는다.
237       preintegrated->integrateMeasurement(imu.head<3>(), imu.tail<3>(), dt);
```



$$\Delta\tilde{\mathrm{R}}_{ij}(\mathbf{b}_i^g) \simeq \Delta\tilde{\mathrm{R}}_{ij}(\bar{\mathbf{b}}_i^g)\,\mathrm{Exp}\left(\frac{\partial\Delta\bar{\mathrm{R}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}^g\right) \tag{44}$$

$$\Delta\tilde{\mathbf{v}}_{ij}(\mathbf{b}_i^g,\mathbf{b}_i^a) \simeq \Delta\tilde{\mathbf{v}}_{ij}(\bar{\mathbf{b}}_i^g,\bar{\mathbf{b}}_i^a) + \frac{\partial\Delta\bar{\mathbf{v}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}_i^g + \frac{\partial\Delta\bar{\mathbf{v}}_{ij}}{\partial\mathbf{b}^a}\delta\mathbf{b}_i^a$$

$$\Delta\tilde{\mathbf{p}}_{ij}(\mathbf{b}_i^g,\mathbf{b}_i^a) \simeq \Delta\tilde{\mathbf{p}}_{ij}(\bar{\mathbf{b}}_i^g,\bar{\mathbf{b}}_i^a) + \frac{\partial\Delta\bar{\mathbf{p}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}_i^g + \frac{\partial\Delta\bar{\mathbf{p}}_{ij}}{\partial\mathbf{b}^a}\delta\mathbf{b}_i^a.$$

IMU measurement integration (preintegrated measurement model) C. Incorporating Bias Updates Appendix B. Bias Correction via First-Order Updates 참고

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
240       Vector7 gps;
241       for (int i = 0; i < 6; ++i) {
242         getline(file, value, ',');
243         gps(i) = stof(value.c_str());
244       }
245       getline(file, value, '\n');
246       gps(6) = stof(value.c_str());
247
248       correction_count++;
249
250       // Adding IMU factor and GPS factor and optimizing.
251       auto preint_imu =
252           dynamic_cast<const PreintegratedImuMeasurements&>(*preintegrated);
253
254       // ImuFactor : the previous pose and velocity,
255       // the current pose and velocity, and the current IMU bias
256       ImuFactor imu_factor(X(correction_count - 1), V(correction_count - 1),
257                            X(correction_count), V(correction_count),
258                            B(correction_count - 1), preint_imu);
259       graph->add(imu_factor);
260       imuBias::ConstantBias zero_bias(Vector3(0, 0, 0), Vector3(0, 0, 0));
261
262       graph->add(BetweenFactor<imuBias::ConstantBias>(
263           B(correction_count - 1), B(correction_count), zero_bias,
264           bias_noise_model));
265
```

Input data 파일에서 GPS data 저장 (Position)

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
240     Vector7 gps;
241     for (int i = 0; i < 6; ++i) {
242       getline(file, value, ',');
243       gps(i) = stof(value.c_str());
244     }
245     getline(file, value, '\n');
246     gps(6) = stof(value.c_str());
247
248     correction_count++;
249
250     // Adding IMU factor and GPS factor and optimizing.
251     auto preint_imu =
252         dynamic_cast<const PreintegratedImuMeasurements&>(*preintegrated);
253
254     // ImuFactor : the previous pose and velocity,
255     // the current pose and velocity, and the current IMU bias
256     ImuFactor imu_factor(X(correction_count - 1), V(correction_count - 1),
257                          X(correction_count), V(correction_count),
258                          B(correction_count - 1), preint_imu);
259     graph->add(imu_factor);
260     imuBias::ConstantBias zero_bias(Vector3(0, 0, 0), Vector3(0, 0, 0));
261
262     graph->add(BetweenFactor<imuBias::ConstantBias>(
263         B(correction_count - 1), B(correction_count), zero_bias,
264         bias_noise_model));
265
```

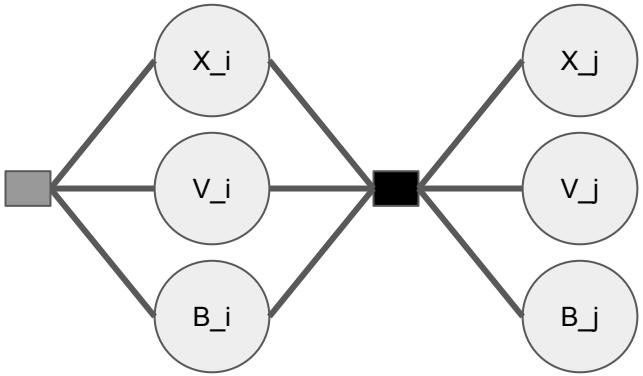
IMU Factor의 초기화

- 이전 state Pose (i)
- 이전 state Velocity (i)
- 현재 state Pose (j)
- 현재 state Velocity (j)
- 이전 state Bias (i)
- ij 사이의 Preintegration 값

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
240       Vector7 gps;
241       for (int i = 0; i < 6; ++i) {
242         getline(file, value, ',');
243         gps(i) = stof(value.c_str());
244       }
245       getline(file, value, '\n');
246       gps(6) = stof(value.c_str());
247
248       correction_count++;
249
250       // Adding IMU factor and GPS factor and optimizing.
251       auto preint_imu =
252           dynamic_cast<const PreintegratedImuMeasurements&>(*preintegrated);
253
254       // ImuFactor : the previous pose and velocity,
255       // the current pose and velocity, and the current IMU bias
256       ImuFactor imu_factor(X(correction_count - 1), V(correction_count - 1),
257                            X(correction_count), V(correction_count),
258                            B(correction_count - 1), preint_imu);
259       graph->add(imu_factor);
260       imuBias::ConstantBias zero_bias(Vector3(0, 0, 0), Vector3(0, 0, 0));
261
262       graph->add(BetweenFactor<imuBias::ConstantBias>(
263           B(correction_count - 1), B(correction_count), zero_bias,
264           bias_noise_model));
265
```



Preintegrated IMU factor를 Graph에 추가

Equation (26)

$$\mathcal{X}_k^\star \doteq \arg\min_{\mathcal{X}_k} -\log_e p(\mathcal{X}_k|\mathcal{Z}_k)$$

$$= \arg\min_{\mathcal{X}_k} \|\mathbf{r}_0\|^2_{\Sigma_0} + \sum_{(i,j)\in\mathcal{K}_k} \|\mathbf{r}_{\mathcal{I}_{ij}}\|^2_{\Sigma_{ij}}$$

$$\mathbf{r}_{\Delta \mathrm{R}_{ij}} \doteq \mathrm{Log}\left(\left(\Delta\tilde{\mathrm{R}}_{ij}(\bar{\mathbf{b}}_i^g)\mathrm{Exp}\left(\frac{\partial\Delta\bar{\mathrm{R}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}^g\right)\right)^\mathsf{T}\mathrm{R}_i^\mathsf{T}\mathrm{R}_j\right)$$

$$\mathbf{r}_{\Delta \mathbf{v}_{ij}} \doteq \mathrm{R}_i^\mathsf{T}(\mathbf{v}_j - \mathbf{v}_i - \mathbf{g}\Delta t_{ij})$$

$$- \left[\Delta\tilde{\mathbf{v}}_{ij}(\bar{\mathbf{b}}_i^g, \bar{\mathbf{b}}_i^a) + \frac{\partial\Delta\bar{\mathbf{v}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}^g + \frac{\partial\Delta\bar{\mathbf{v}}_{ij}}{\partial\mathbf{b}^a}\delta\mathbf{b}^a\right]$$

$$\mathbf{r}_{\Delta \mathbf{p}_{ij}} \doteq \mathrm{R}_i^\mathsf{T}\left(\mathbf{p}_j - \mathbf{p}_i - \mathbf{v}_i\Delta t_{ij} - \frac{1}{2}\mathbf{g}\Delta t_{ij}^2\right)$$

$$- \left[\Delta\tilde{\mathbf{p}}_{ij}(\bar{\mathbf{b}}_i^g, \bar{\mathbf{b}}_i^a) + \frac{\partial\Delta\bar{\mathbf{p}}_{ij}}{\partial\mathbf{b}^g}\delta\mathbf{b}^g + \frac{\partial\Delta\bar{\mathbf{p}}_{ij}}{\partial\mathbf{b}_a}\delta\mathbf{b}^a\right] \quad (45)$$

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
240       Vector7 gps;
241       for (int i = 0; i < 6; ++i) {
242         getline(file, value, ',');
243         gps(i) = stof(value.c_str());
244       }
245       getline(file, value, '\n');
246       gps(6) = stof(value.c_str());
247
248       correction_count++;
249
250       // Adding IMU factor and GPS factor and optimizing.
251       auto preint_imu =
252           dynamic_cast<const PreintegratedImuMeasurements&>(*preintegrated);
253
254       // ImuFactor : the previous pose and velocity,
255       // the current pose and velocity, and the current IMU bias
256       ImuFactor imu_factor(X(correction_count - 1), V(correction_count - 1),
257                            X(correction_count), V(correction_count),
258                            B(correction_count - 1), preint_imu);
259       graph->add(imu_factor);
260       imuBias::ConstantBias zero_bias(Vector3(0, 0, 0), Vector3(0, 0, 0));
261
262       graph->add(BetweenFactor<imuBias::ConstantBias>(
263           B(correction_count - 1), B(correction_count), zero_bias,
264           bias_noise_model));
265
```

zero bias를 가정하고 Combined IMU factor를 추가

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
240       Vector7 gps;
241       for (int i = 0; i < 6; ++i) {
242         getline(file, value, ',');
243         gps(i) = stof(value.c_str());
244       }
245       getline(file, value, '\n');
246       gps(6) = stof(value.c_str());
247
248       correction_count++;
249
250       // Adding IMU factor and GPS factor and optimizing.
251       auto preint_imu =
252           dynamic_cast<const PreintegratedImuMeasurements&>(*preintegrated);
253
254       // ImuFactor : the previous pose and velocity,
255       // the current pose and velocity, and the current IMU bias
256       ImuFactor imu_factor(X(correction_count - 1), V(correction_count - 1),
257                            X(correction_count), V(correction_count),
258                            B(correction_count - 1), preint_imu);
259       graph->add(imu_factor);
260       imuBias::ConstantBias zero_bias(Vector3(0, 0, 0), Vector3(0, 0, 0));
261
262       graph->add(BetweenFactor<imuBias::ConstantBias>(
263           B(correction_count - 1), B(correction_count), zero_bias,
264           bias_noise_model));
265
```

수행 이유? (ImuFactor.pdf Combined IMU Factor 참고)

- 위의 IMU factor는 각각 state에 대해 bias modeling을 독립적으로 모델링할 것을 요구
- 그러나 Preintegration을 수행할 시, bias간에 상관관계가 있고 이에 의존

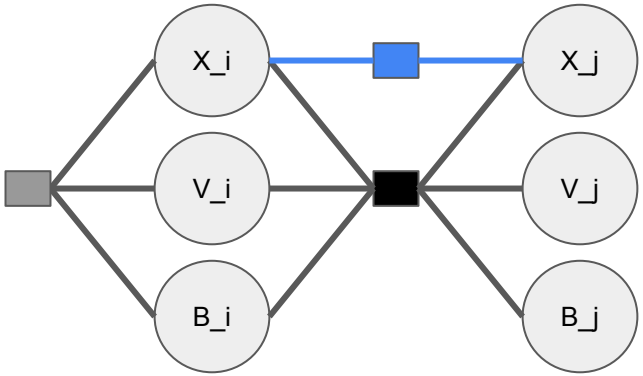
zero bias를 가정하고 Combined IMU fator를 추가

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
266     auto correction_noise = noiseModel::Isotropic::Sigma(3, 1.0);
267     GPSFactor gps_factor(X(correction_count),
268                          Point3(gps(0),   // N,
269                                 gps(1),   // E,
270                                 gps(2)),  // D,
271                          correction_noise);
272     graph->add(gps_factor);
273
274     // Now optimize and compare results.
275     // 이전 imu에서 쌓은 preintegrated를 이용해서 현재 imu의 state를 예측한다.
276     prop_state = preintegrated->predict(prev_state, prev_bias);
277
278     // 예측한 state를 initial_values에 넣는다.
279     initial_values.insert(X(correction_count), prop_state.pose());
280     initial_values.insert(V(correction_count), prop_state.v());
281     initial_values.insert(B(correction_count), prev_bias);
```

GPS factor를 Graph에 추가



IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
266     auto correction_noise = noiseModel::Isotropic::Sigma(3, 1.0);
267     GPSFactor gps_factor(X(correction_count),
268                          Point3(gps(0),   // N,
269                                 gps(1),   // E,
270                                 gps(2)),  // D,
271                          correction_noise);
272     graph->add(gps_factor);
273
274     // Now optimize and compare results.
275     // 이전 imu에서 쌓은 preintegrated를 이용해서 현재 imu의 state를 예측한다.
276     prop_state = preintegrated->predict(prev_state, prev_bias);
277
278     // 예측한 state를 initial_values에 넣는다.
279     initial_values.insert(X(correction_count), prop_state.pose());
280     initial_values.insert(V(correction_count), prop_state.v());
281     initial_values.insert(B(correction_count), prev_bias);
```

j번째 state를 predict

IMUFactorExample code review

- Factor 추가 (IMU preintegration factor, GPS factor)

```
266     auto correction_noise = noiseModel::Isotropic::Sigma(3, 1.0);
267     GPSFactor gps_factor(X(correction_count),
268                          Point3(gps(0),   // N,
269                                 gps(1),   // E,
270                                 gps(2)),  // D,
271                          correction_noise);
272     graph->add(gps_factor);
273
274     // Now optimize and compare results.
275     // 이전 imu에서 쌓은 preintegrated를 이용해서 현재 imu의 state를 예측한다.
276     prop_state = preintegrated->predict(prev_state, prev_bias);
277
278     // 예측한 state를 initial_values에 넣는다.
279     initial_values.insert(X(correction_count), prop_state.pose());
280     initial_values.insert(V(correction_count), prop_state.v());
281     initial_values.insert(B(correction_count), prev_bias);
```

예측한 state를 value값으로 저장

IMUFactorExample code review

- ISAM을 활용한 최적화 수행

```
283     Values result;
284
285     if (use_isam) {
286       isam2->update(*graph, initial_values);
287       // isam2->update();
288       result = isam2->calculateEstimate();
289
290       // reset the graph
291       graph->resize(0);
292       initial_values.clear();
293
294     } else {
295       LevenbergMarquardtOptimizer optimizer(*graph, initial_values);
296       result = optimizer.optimize();
297     }
```

$$\hat{x} \leftarrow \mathcal{R}_{\hat{x}}(\delta x^{\star}).$$

ISAM을 활용한 factor graph 최적화

Equation (26)

$$\mathcal{X}_k^{\star} \doteq \arg\min_{\mathcal{X}_k} -\log_e p(\mathcal{X}_k|\mathcal{Z}_k)$$

$$= \arg\min_{\mathcal{X}_k} \|\mathbf{r}_0\|^2_{\mathbf{\Sigma}_0} + \sum_{(i,j)\in\mathcal{K}_k} \|\mathbf{r}_{\mathcal{I}_{ij}}\|^2_{\mathbf{\Sigma}_{ij}}.$$

IMUFactorExample code review

- 결과 비교를 위한 과정 - (GT : GPS값으로 만든 state, current : 예측한 state값)

```
307       // Print out the position and orientation error for comparison.
308       Vector3 gtsam_position = prev_state.pose().translation();
309       Vector3 position_error = gtsam_position - gps.head<3>();
310       current_position_error = position_error.norm();
311
312       Quaternion gtsam_quat = prev_state.pose().rotation().toQuaternion();
313       Quaternion gps_quat(gps(6), gps(3), gps(4), gps(5));
314       Quaternion quat_error = gtsam_quat * gps_quat.inverse();
315       quat_error.normalize();
316       Vector3 euler_angle_error(quat_error.x() * 2, quat_error.y() * 2,
317                                 quat_error.z() * 2);
318       current_orientation_error = euler_angle_error.norm();
319
320       // display statistics
321       cout << "Position error:" << current_position_error << "\t "
322            << "Angular error:" << current_orientation_error << "\n";
323
324       fprintf(fp_out, "%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f,%f\n",
325               output_time, gtsam_position(0), gtsam_position(1),
326               gtsam_position(2), gtsam_quat.x(), gtsam_quat.y(), gtsam_quat.z(),
327               gtsam_quat.w(), gps(0), gps(1), gps(2), gps_quat.x(),
328               gps_quat.y(), gps_quat.z(), gps_quat.w());
329
330       output_time += 1.0;
331
332     } else {
333       cerr << "ERROR parsing file\n";
334       return 1;
335     }
336   }
```

IMUFactorExample code review

- Result

```
Position error:0.0580787        Angular error:0.000344909
Position error:0.0542306        Angular error:0.00043091
Position error:0.0491488        Angular error:0.00117964
Position error:0.0431538        Angular error:0.00187845
Position error:0.0365589        Angular error:0.00250925
Position error:0.0296588        Angular error:0.0030587
Position error:0.0227248        Angular error:0.00351803
Position error:0.0159968        Angular error:0.00388278
Position error:0.00967761       Angular error:0.0041524
Position error:0.00393953       Angular error:0.00432974
Position error:0.00116716       Angular error:0.00442057
Position error:0.00540293       Angular error:0.00443294
Position error:0.00883039       Angular error:0.00437666
Position error:0.0114181        Angular error:0.00426265
Position error:0.0131907        Angular error:0.00410246
Position error:0.0141875        Angular error:0.00390771
Position error:0.0144784        Angular error:0.00368972
Position error:0.0141501        Angular error:0.00345906
Position error:0.0133007        Angular error:0.00322532
Position error:0.0120347        Angular error:0.00299688
Position error:0.0104587        Angular error:0.00278072
Position error:0.00867011       Angular error:0.00258241
Position error:0.00678414       Angular error:0.00240605
Position error:0.00486084       Angular error:0.00225433
Position error:0.00302052       Angular error:0.00212862
Position error:0.00127799       Angular error:0.00202912
Position error:0.000365291      Angular error:0.00195496
Position error:0.00165817       Angular error:0.00190445
Position error:0.00277708       Angular error:0.00187521
Position error:0.0036316        Angular error:0.00186436
Position error:0.00425051       Angular error:0.00186875
Position error:0.00462199       Angular error:0.00188507
Position error:0.00478143       Angular error:0.00191003
Position error:0.00473683       Angular error:0.00194049
Complete, results written to imuFactorExampleResults.csv
```

IMUFactorExample code review

- Result